
MAGAZINE
MSX
AÑO II
Núm. 18
Nov. 1986
300 Ptas.
Unidad de discos
ML-30 FD
Test:
interface RS-232 C
Utilización de ficheros
El BIOS
de la memoria
de video
DRIVE 1
DRIVE 2
MITSUBISHI
POWER
ML-30FD
LOS DISKETTES AL DESCUBIERTO

DIRECTOR:
Juan Arencibia.
COLABORADORES:
Octavio López, Angel Zarazaga, Teresa Aranda, Ricardo García.
DISEÑO:
Benito Gil
Editada por:
PUBLINFORMATICA, S.A.
C/ Bravo Murillo, 377 - 5.º A
Tel.: 733 74 13
28020 Madrid.
Telex 48877 OPZXE
PRESIDENTE:
Fernando Bolín.
DIRECTOR EDITORIAL REVISTAS DE USUARIOS:
Juan Arencibia.
DIRECTOR DE VENTAS:
Antonio González.
JEFE DE PRODUCCIÓN:
Miguel Onieva.
SERVICIO AL CLIENTE:
Julia González.
Tel.: 733 79 69
DIRECCION, REDACCION Y ADMINISTRACIÓN:
C/ Bravo Murillo, 377 - 5.º A.
Tel.: 733 74 13
28020 Madrid.
PUBLICIDAD EN MADRID:
Emilio García.
PUBLICIDAD EN BARCELONA:
Lidia Cendros.
C/ Pelayo, 12.
Tel.: (93) 301 47 00 Ext. 27-28.
08001 Barcelona.
Depósito Legal: M. 16.755-1985
Impreso en Héroes, S.A.
C/ Torrelara, 8. 28016 Madrid.
Distribuye:
S.G.E.L. Avda. Valdelaparra, s/n.
Alcobendas (Madrid).
DISTRIBUIDORES:
VENEZUELA: SIPAM, S.A.
Avda. República Dominicana, 541
ARGENTINA: DISTRIBUIDORA INTERCONTINENTAL BUENOS AIRES.

El P.V.P. para Ceuta, Melilla y Canarias, incluido servicio aéreo será de 300 ptas. sin I.V.A.

SUSCRIPCIONES:
Rogamos dirija toda la correspondencia relacionada con suscripciones a:
MSX
EDISA: Tel, 415 97 12
C/López de Hoyos, 141-5.º
28002 MADRID
(Para todos los pagos reseñar solamente MSX)
Para la compra de ejemplares atrasados diríjanse a la propia editorial
MSX
C/Bravo Murillo, 377-5.º A
Tel. 733 74 13 28020 MADRID

Si deseas colaborar en MSX remite tus artículos o programas a Bravo Murillo 377, 5.º A. 28020 Madrid. Los programas deberán estar grabados en cassette y los artículos mecanografiados.
A efectos de remuneración, se analiza cada colaboración aisladamente, estudiando su complejidad y calidad.



Este mes se celebra en Madrid SIMO'86. A simple vista, este hecho no reviste de mayor importancia, si no fuera por que va a ser el primero que se celebre después de la entrada de España en las Comunidades Europeas. Del 14 al 21 de este mes se celebrará, en el recinto de IFEMA, esta importante feria que contará con la participación de 1.200 expositores de 23 países.

Contará con cuatro zonas especiales que serán: IN-PROSIMO, SIMOMICRO, SIMOLOG y SIMODAT, cada una de las cuales con un cometido específico. El primero tiene como misión la de facilitar al profesional una visita útil y eficaz. En SIMOMICRO, se exhibirán desde ordenadores domésticos, hasta ordenadores personales orientados hacia la pequeña y mediana empresa.

SIMOLOG presentará programas de aplicación, sistemas de explotación y lenguajes de programación, mostrando a su vez, los avances del logicial en este campo. Y en SIMODAT, podremos comprobar, la creciente industria de las BASES DE DATOS con sus productores y distribuidores.

En suma, el SIMO'86 se presenta como una ventana abierta al futuro, donde se centrará la atención en los avances realizados por parte de aquellas empresas que han cambiado el ordenador doméstico por el personal. Por otro lado, este mes de noviembre, ha traído a nuestras páginas un tema de portada sumamente interesante, como es el diskette. Las unidades de discos de 3.5 pulgadas están debancando, lenta pero inexorablemente, al eterno disco de 5.25 pulgadas. Primero fueron los fabricantes de MSX, que en un gran esfuerzo unificaron criterios sobre este periférico, luegon aparecieron los IBM PC con dichas unidades, ahora existen algunos compatibles PC (como el Toshiba T1100) que lo incorpora... poco a poco se está asentando en los ordenadores pequeños/medianos, como ocurrió en su momento con los discos de 8 pulgadas. ¿A qué es debido semejante cambio? Muchos son los factores que han influido en su creciente desarrollo, pero no vamos a contar más. En las páginas siguientes, pusimos a los diskettes bajo la lupa.





6

Noticias. Nuevos programas de ERBE. Investrónica a la conquista del mercado de PC y compatibles. Sony y sus próximos productos.

8

Los diskettes bajo la lupa. Este importante periférico cuenta con las simpatías de todos los usuarios. Sin embargo, pocos son los que verdaderamente saben cómo funciona.

16

Software. Buenos y entretenidos programas para todos los gustos. Este mes traemos Master of the Lamps, BOOM!, Invierte y Gana, Calculator New, Hopper y Oh Shit!

24

Síntesis de voz para MSX (y II). Continuación del interesante artículo sobre síntesis de voz, muy útil para incorporar a nuestros programas.

30

Libros. Computación Interactiva. Sistemas. Diálogos. Orientado hacia la comunicación entre ordenador y el usuario. Inteligencia Artificial y Sistemas Expertos. Una breve descripción sobre este importante y siempre interesante tema.



mario

32

Test: RS-232C. El complemento ideal para cualquier ordenador es el interface RS-232C. Con él podremos conectarlo a todos los periféricos imaginables.

36

Utilización de ficheros de datos. Las aplicaciones sobre tratamientos de ficheros son muy importantes, sobre todo cuando se cuenta con una unidad de discos, elemento fundamental en este tipo de programas.

40

BASIC en C/M. Cómo realizar programas en BASIC aprovechando la potencia y rapidez del Código Máquina.



44

El BIOS de la Memoria de Vídeo. O cómo aprovechar todas las posibilidades gráficas de los ordenadores MSX. El BIOS es el núcleo de la Memoria de Vídeo, con él, llegaremos a dominar todo lo referente a los gráficos.

52

SVI 318/328, SCREEN 2. Gráficos de baja resolución. El inesperado éxito de esta sección nos ha obligado a continuar con el tema. En esta ocasión tratamos el modo SCREEN 2.

58

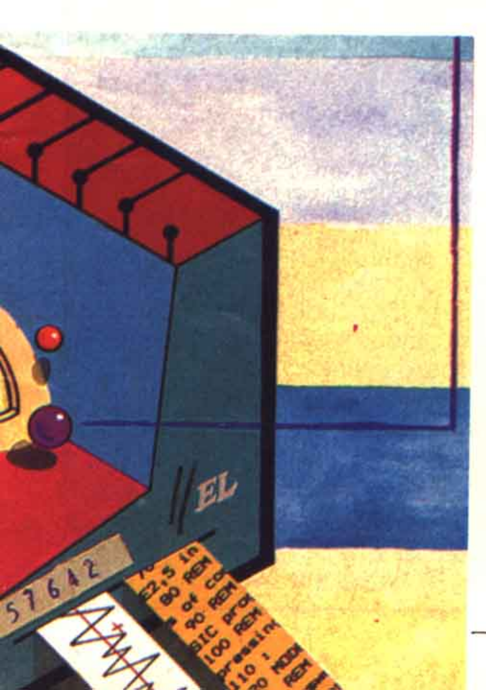
Programa: Biblioteca. Interesante programa donde el lector comprobará la facilidad de crear ficheros con una utilidad específica.

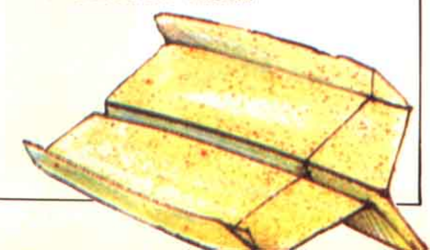
65

Compro, Vendo. Hemos cambiado un poco esta sección y facilitado la labor de nuestros lectores para que se pongan en contacto con la revista y efectúen sus intercambios.

66

Rincón del lector. Donde todas vuestras dudas encontrarán la solución ideal.







Investrónica, ¿ordenadores o música?

Investrónica, S.A., empresa líder en alta tecnología aplicada a la industria textil, informática, etc., y uno de los mayores distribuidores de ordenadores personales del mercado español, se ha introducido en el área de los compatibles con la gama INVES PC y en el mundo del sonido con la línea INVES.

La gama de compatibles consta de tres configuraciones de distintas capacidades y prestaciones: INVES PC 256X, INVES PC 640A e INVES PC 640X, para adaptarse a las necesidades empresariales o personales, tanto de trabajo como de presupuesto.

Por otro lado, la línea de sonido INVES, se inicia con la cadena 100 HF con giradiscos incorporado. El equipo se amplía con el sintonizador digital y compact disc CD 300 HF y por último el Compact Disc CD 200.

La novedad de todos los productos de Investrónica, S.A., radica en que posee una de las mejores relaciones calidad/precio del mercado, siendo su punto fuerte las prestaciones.

Idealogic en la vanguardia de la enseñanza

Desde sus inicios Idealogic, S.A., ha sido una empresa puntera en el desarrollo y fomento de las aplicaciones de la informática en sus más amplias posibilidades y con especial interés en el ámbito educativo.

Centre Logo representa uno de los puntales de esta empresa para el ámbito de la informática aplicada a la educación. Desde él, se pretende acercar al educador a las tecnologías y metodologías que requiere la utilización de los ordenadores en un centro educativo.

El Centre Logo colabora desde sus inicios con entidades Públicas y privadas, como el Gobierno Andorrano, la Federación de Escuelas Religiosas, y muchos otros. Las experiencias recogidas en este espacio de tiempo y el interés despertado, obligará a Idealogic, S.A., a extender sus actividades a otros lugares como Madrid, Almería, Zaragoza y Vigo.

El lanzamiento de, Omnibus, Micro Torno, PROFIL, MANU 3, FAO 2, Tarjeta RAM y Tarjeta de Relés, van a ser elementos muy útiles para los medios educativos, profesionales e incluso los hobbistas en muy diversos campos (electrónica, diseño, robótica, mecánica,...).

Para más información dirigirse a:

Idealogic, S.A.
Valencia, 85
08029 Barcelona

Centre Logo
Entanza, 218
08029 Barcelona



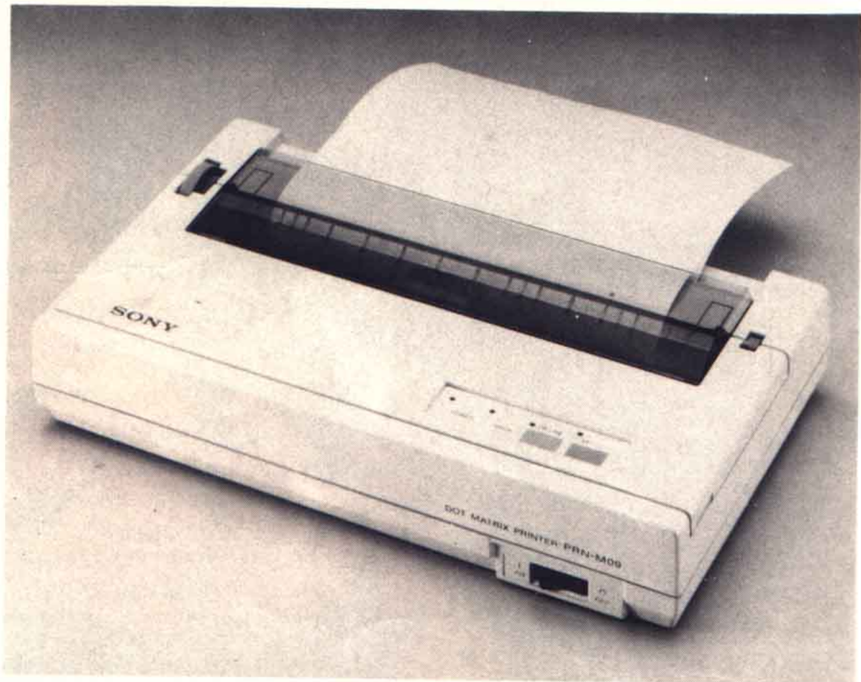
Nuevos equipos Sony

Durante el otoño, Sony lanzará al mercado español sus nuevos productos. Entre ellos podemos destacar la nueva generación de ordenadores MSX-II y los nuevos periféricos.

En lo que a ordenadores se refiere, el nuevo HB-F700S, va a situarse por encima del HG-500P. Ello se debe a la incorporación de un ratón, una unidad de discos de 1Mb (720K formateados) y diversos programas de aplicación. Estos constan de una Hoja de Cálculo, un Procesador de Textos, una Base de Datos y Gráficas de Gestión. Además cuenta con todas las importantes cualidades del mencionado HB-500P, tales como teclado independiente, unidad central con todos los conectores, etc.

A un nivel inferior se encuentra el HB-F9P. La única diferencia existente entre éste y el anterior, radica en que no lleva unidad de discos incorporada, pero posee todas las importantes características de los ordenadores de la II Generación. Esta máquina, orientada hacia el pequeño/mediano usuario, es la opción idónea para iniciarse en el tema de la informática. Al contrario que el anterior, el software viene incorporado en ROM y consta de un Listín Telefónico, una Agenda Personal, un reloj, calendario y calculadora. En cuanto a periféricos, cabe señalar el importante paso que se ha dado. Con la aparición del SDC-600S grabador/reproductor de cassette de alta velocidad, se ha llegado a obtener algo que parecía difícil si no imposible, un cassette especialmente preparado para ordenador (y no esos aparatos que dicen que lo son, sólo para comprobar que a los tres días ya no funcionan). Dispone de dos velocidades de reprodución, además de un selector de fase para evitar los problemas de carga.

El HBI-232 cartucho de comunicaciones, permite conectar los HIT-BIT con cualquier ordenador que posea el interface RS-232C, bien directamente o mediante un modem.



La impresora matricial PRN-MO9 tiene una velocidad de impresión de 75 caracteres/seg. Dispone de tres tamaños de letra pudiéndose imprimir en alta calidad (NLQ).

Por último, lo que todos estábamos esperando, el joystick profesional (JS-70). Su característica principal es una palanca corta y fuerte, similar a las que hay en las máquinas de arcade, dispuesta en una carcasa sólida y ancha con dos pulsadores que permiten una desenvoltura total para diestros y zurdos.

Nuevos programas de ERBE

ERBE empresa distribuidora de programas de casas tan conocidas como Melbourne House, Ocean... va a lanzar al mercado dos programas muy buenos, además de un paquete compuesto por los diez éxitos del año a un precio que hará temblar y tambalear el negocio de los piratas.

Los dos juegos en cuestión son: Las Tres Luces de Glaurung y B.C. II, Grog's Revenge. Ambos son entretenidos y muy buenos, aunque por razones de tiempo no hemos podido comentarlos para este mes. Lo haremos el mes que viene en nuestras páginas habituales.

Por otro lado, la aparición de un paquete de 10 programas que han sido éxito durante el presente año, a un precio de 3.995 ptas., significa un claro empuje por parte de ERBE para contrarestrar el negocio pirata. El paquete consta de los programas siguientes: Showjumper, Alien-8, Knight Lore, Gunfright, Nightshade, Jack the Nipper, Valkyer, Bounder, Mapgame y Jet Set Willy II.





Los diskettes bajo la lupa

Un examen exhaustivo de las características del medio de almacenamiento masivo ideal para ordenadores domésticos.

Cualquier usuario que haya sufrido las incomodidades del uso de una cinta cassette para guardar programas u otros datos sentirá una inmediata satisfacción la primera vez que use un disco. En efecto, la rapidez y simpleza de manejo de este elemento contrastan de forma espectacular con la lentitud del cassette. Recurriendo a los números, si con la cinta las velocidades de transmisión de datos son de 1.200 ó 2.400 *baudios* (bits por segundo), con el disco pasamos a tener la «astronómica» velocidad de 250.000 *baudios;* esto quiere decir que un programa de 16 Kbytes tarda de 1 a 2 minutos en cargar con un cassette, mientras que usando el disco tardaría sólo ¡1 segundo!

Existe otra ventaja importante en el uso del diskette: es, al contrario de la cinta, un soporte FORMATEADO. El *formateo* es una operación que se efectúa antes de usar por primera vez un disco virgen (o sea, sin nada grabado). En ella se graban sobre la superficie del disco una serie de señales magnéticas que servirán, en su momento, como referencia para localizar los ficheros grabados; con esto se solucionan dos importantes problemas: localizar un determinado fichero, sin tener que pasar antes por todos los demás como ocurre con la cinta, y tener la seguridad de que al grabar un nuevo fichero no lo hacemos sobre otro que nos interesa conservar.

El único inconveniente de los discos es su precio, relativamente alto, y el de los dispositivos *(«drives»)* necesarios para usarlos. Es-



Foto 1. Discos de 3.5 pulgadas de simple y doble cara.

te precio está justificado en que los sistemas de disco son muy específicos, ya que sólo pueden usarse en ordenadores. Por otra parte, la sencillez de manejo del disco oculta el complejo mundo de los detalles propios del *diskette,* así como de los dispositivos físicos (controlador, *drive*) y lógicos (sistemas operativos) que nos permiten usarlo.

Tipos de diskettes

El diskette (llamado también *floppy disk* o disco flexible) es una lámina circular de *mylar* (material plástico), recubierta de un óxido que le da una superficie similar a la de una cinta de grabación magnética. El disco va encerrado en una funda que lo protege de suciedad y deformaciones. El disco no se saca nunca de esta funda; ésta lleva una serie de aberturas que permiten hacer girar el disco y leer su superficie.

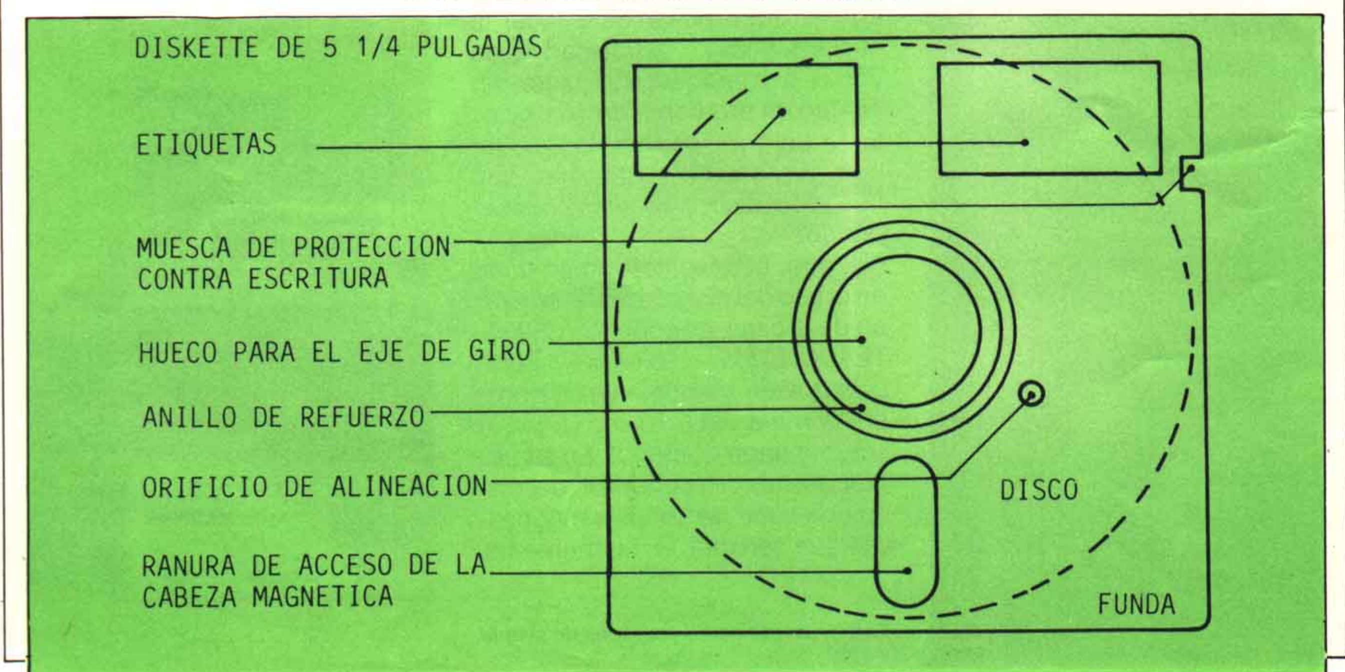
El tamaño de los discos no se encuentra totalmente normalizado. Los más corrientes son los de 8, 5 1/4, 3 1/2 y 3 pulgadas. Estos formatos son físicamente incompatibles entre sí, ya que los *drives* se hacen a la medida del *diskette.*

La compatibilidad viene dada

TAMAÑO	3 1/2 pulgadas			
FORMATO	891	892	881	882
ENTRADAS A DIRECTORIO	112	112	112	112
FATID (BYTE IDENTIFICADOR)	&HØF8	&HØF9	&HØFA	&HØFB
NUMERO DE CARAS	1	2	1	2
PISTAS POR CARA	8Ø	8Ø	8Ø	8Ø
SECTORES POR PISTA	9	9	8	8
BYTES POR SECTOR	512	512	512	512
SECTORES POR CLUSTER	2	2	2	2
SECTORES POR FAT	2	3	1	2
NUMERO DE FATS	2	2	2	2
CAPACIDAD TOTAL (BYTES)	368.64Ø	737.28Ø	327.68Ø	655.36Ø

Figura 1. Características de los discos de 3.5 pulgadas.

Figura 2. Descripción del disco de 5.25 pulgadas.



TAMAÑO	5 1/4 pulgadas			
FORMATO	491	492	481	482
ENTRADAS A DIRECTORIO	64	112	64	112
FATID (BYTE IDENTIFICADOR)	&HØFC	&HØFD	&HØFE	&HØFF
NUMERO DE CARAS	1	2	1	2
PISTAS POR CARA	4Ø	4Ø	4Ø	4Ø
SECTORES POR PISTA	9	9	8	8
BYTES POR SECTOR	512	512	512	512
SECTORES POR CLUSTER	1	2	1	2
SECTORES POR FAT	1	2	1	2
NUMERO DE FATS	2	2	2	2
CAPACIDAD TOTAL (BYTES)	184.32Ø	368.64Ø	163.84Ø	327.68Ø

Figura 3. Características de los discos de 5.25 pulgadas.

no sólo por el tamaño, sino también por la forma en que se graban los datos; esta característica depende de la electrónica del interface (o controlador) y del sistema operativo.

Los *diskettes* pueden tener una o dos caras, dependiendo de que el drive tenga una o dos cabezas magnéticas. Los *drives* de dos cabezas pueden usar indistintamente *diskettes* de una o dos caras, sin embargo un *drive* de una sola cabeza sólo podrá leer, evidentemente, discos de una sola cara; esto es lo que suele llamarse «compatibilidad ascendente».

El último aspecto característico de los discos es la «densidad de grabación», que es un indicador de cuanta capacidad puede tener físicamente el disco. Puede ser simple o doble.

MSX y los diskettes

El criterio aplicado al ampliar la norma MSX a los *diskettes* es muy liberal, podríamos incluso decir que en exceso. Se aceptan dos tamaños (5 1/4 y 3 1/2 pulgadas) en versión de una o dos caras. Para colmo, por cada uno de estos formatos físicos hay 2 tipos de posibles de formato magnético; esto da 8 posibles combinaciones. Esta diversidad tiene la ventaja de ofrecer una gama de precios más amplia, pero falla a la hora de la total compatibilidad. Para asegurarnos de que un disco es totalmente compatible con nuestro sistema debemos comprobar que tiene el tamaño apropiado y que su número de caras es igual o menor que el correspondiente a nuestro drive.

Figura 4. Descripción del disco de 3.5 pulgadas.

DISKETTE DE 3 1/2"
CARA SUPERIOR
LENGUETA DE PROTECCION
RANURA DE ACCESO
ETIQUETA
EJE DE GIRO
CARA INFERIOR
ORIFICIO DE PROTECCION CONTRA ESCRITURA
PESTAÑA DESLIZANTE

La tendencia actual apunta al uso generalizado del formato de 3 1/2 pulgadas, ya que ocupan menos espacio, tienen una capacidad muy grande y son más manejables por estar totalmente protegidos dentro de una funda de plástico rígido. La mayor parte de los *drives* existentes en el mercado corresponden a este formato (el único *drive* de 5 1/4" disponible, de momento, es el Spectravídeo SVI-707).



Foto 2. Unidad de discos de Mitsubishi ML-30FD, un potente periférico.

Funcionamiento del diskette

Para leer o grabar información se hace girar el disco a gran velocidad (300 r.p.m.). Los datos se transmiten a través de una cabeza magnética que recorre pistas *(tracks)* concéntricas sobre la superficie del disco; la cabeza puede moverse en sentido radial, con lo que pueden recorrerse las dos dimensiones de dicha superficie. Es importante hacer notar el hecho de que, a diferencia de los discos sonoros en los que la aguja traza una espiral continua, en los *diskettes* la cabeza salta radialmente al completar una pista.

Las pistas se dividen en sectores; la división en pistas y sectores se hace durante el *formateo.* En los sistemas MSX cada pista tiene 8 ó 9 sectores de 512 *bytes.* Un sector es la cantidad mínima de información que se puede leer o escribir en el disco; esto se debe al uso de una técnica llamada DMA (*Direct Memory Access* o Acceso Directo a Memoria). En ella, el sistema usa un área de *RAM* donde recoge o deposita la información transferida desde o hacia el disco; el controlador tratará esta información como un todo. Este sistema tiene la desventaja de desaprovechar espacio en *RAM* y en el diskette, pero aumenta enormemente la velocidad con que se transfieren los datos.

Organización de un diskette

La forma en que trabaja el *diskette* es muy diferente, desde el punto de vista del *software,* a la del cassette. Simplificando conceptos podemos decir que el disco trabaja en dos dimensiones, mientras que la cinta lo hace en una. Hemos visto cómo el ordeandor ha «parcelado» el disco en sectores; sería muy complicado que el usuario llevase un control de cuál es el contenido de cada sector. Este trabajo lo lleva el Sistema Operativo, manteniendo dos tablas en el disco: la *FAT* y el DIRECTORIO.

El directorio es una lista resumida del contenido del disco. Por cada fichero existente hay una ENTRADA al directorio, donde figuran el nombre del fichero, la fecha en que se grabó, el primer *«cluster»* que ocupa (un *«cluster»* está formado por 2 sectores consecutivos, o sea, 1 *Kbyte*) y su extensión en *bytes.*

Foto 3. Su acabado y la disposición vertical permiten ponerlo en cualquier lugar.



La *FAT* (*File Allocation Table* o Tabla de Asignación de Ficheros) es un mapa numérico del disco. A cada elemento de la *FAT* corresponde un *cluster,* e indica si está lleno, si queda en él algo de espacio o si está vacío. La *FAT* se actualiza cada vez que se graba o borra un fichero.

A los sectores se les asigna un número, comenzando por 0, que es el primer sector de la pista más exterior. Los sectores así numerados se llaman «sectores lógicos». El sector 0 contiene un pequeño programa llamado *«BOOT»* que carga el Sistema Operativo cuando se pone en marcha el ordenador con un diskette en el drive. A continuación vienen las *FAT's* (todos los *diskettes* MSX tienen dos *FAT's*): una ocupa, normalmente, los sectores 1 y 2; la otra el 3 y el 4. La *FAT* funciona de la siguiente forma:

BYTE 0 BYTE 1 BYTE 2

BYTE 3 BYTE 4 FAT 0

FAT 1 FAT 2

Comienzo de la FAT

b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0

b_3 b_2 b_1 b_0 b_{11} b_{10} b_9 b_8

b_{11} b_{10} b_9 b_8 b_7 b_6 b_5 b_4

b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0

b b b b b_{11} b_{10} b_9 b_8

Cada elemento de la *FAT* es un número de 12 *bits* (3 *nibbles*): el primer elemento se forma con el *nibble* bajo del segundo *byte* más el primer *byte,* el segundo con el tercer *byte* más el *nibble* alto del segundo *byte,* el tercer elemento como el primero y así sucesivamente (como se sabe, cada *nibble* es una cifra hexadecimal). Los números así formados nos dirán lo siguiente:

$N_2N_1N_0$ = 000 el *cluster* correspondiente está libre.

$N_2N_1N_0$ = FFF el *cluster* está parcialmente ocupado.

$N_2N_1N_0$ = otro número (este número más uno es el del *cluster*) el *cluster* está lleno.

El primer *byte* de la primera *FAT* (sector 1 del disco) se llama *FATID (FAT INDENTIFIER)* y, según veremos más adelante, sirve para identificar el tipo de formato del disco.

El directorio es el bloque de información que sigue a la *FAT* en el disco. Cada entrada consta de 32 *bytes;* por tanto, por cada sector hay 16 entradas. Excepto los discos de 5 1/4" y una sola cara, que sólo admiten 64 entradas, todos los demás pueden tener hasta 112 entradas al directorio (7 sectores). El formato de cada entrada es el siguiente:

NOMBRE DEL FICHERO

RESERVADOS

HORA FECHA

NO SUELE USARSE

LONGITUD DEL FICHERO EN BYTES

PRIMER CLUSTER OCUPADO POR EL FICHERO

B_0 B_1 B_2 B_3 B_4 B_5 B_6 B_7

B_8 B_9 B_{10} B_{11} B_{12} B_{13}

B_{14} B_{15} B_{16} B_{17} B_{18} B_{19}

B_{20} B_{21} B_{22} B_{23} B_{24} B_{25}

B_{26} B_{27} B_{28} B_{29} B_{30} B_{31}

El nombre de fichero se rellena con espacios hasta totalizar 11 *bytes.* Cuando el primer carácter del nombre es HE5, esto indica que el fichero ha sido borrado; esta operación y el poner a 0 todos los





Foto 4. El controlador tiene todo lo necesario para manejar debidamente una o dos unidades.

punteros correspondientes en la *FAT* es lo único que ocurre en el disco al borrar un fichero; no sería complicado escribir un programa que recuperase ficheros borrados por error (esta función la tienen algunos sistemas operativos, como el MS-DOS en que se denomina *RECOVER*). La longitud de fichero viene dada en el formato *byte* bajo-*byte* alto.

A partir del directorio comienzan los *clusters* que contienen datos. Los punteros de las *FAT* y del directorio indican la posición de cada fichero dentro de esta zona.

En cada operación que se efectúa con el disco se carga en *RAM* una copia de la *FAT*, sea para lectura o para escritura. La transferencia del resto de los datos se efectuará a través del *«buffer» DMA*.

En el siguiente cuadro se resumen todas las características estudiadas. En el apartado «FORMATO», la 1.ª cifra indica el número de pistas (multiplicar por 10), la 2.ª indica el número de sectores por pista y la 3.ª el número de cabezas del *drive* (y, por tanto, el número de caras del disco).

En las capacidades indicadas, que son las que proporciona el formateado, está incluído el espacio que ocupan el programa *BOOT* (que se graba durante la operación de formateo), las *FATs* y el directorio; este espacio es, tomando como ejemplo el formato 891, de 6.144 *bytes*.

Medios software de acceder al disco

Con los ordenadores MSX hay, en principio, dos formas de acceder a la información contenida en diskette: una es el *Disk Basic* y la otra es el MSX-DOS.

El MSX-DOS *(MicroSoft eXtended Disk Operating System)* es lo que puede llamarse un *«interface software»*, es decir, una serie de



Unidad de discos ML-30 FD

Aprovechando la ocasión, vamos a comentar la aparición en nuestro mercado de una unidad de discos de Mitsubishi denominada ML-30FD.

Parece ser que Mitsubishi, está dispuesto a convertirse en uno de los líderes del mercado MSX. Así lo demuestra con sus productos que, reuniendo una gran calidad a un buen acabado, los convierten en la mejor baza a la hora de elegir un equipo. Este es el caso de la unidad de discos con la que vamos a poner punto final a este artículo.

Las características más destacadas de dicha unidad son:

- Usa diskettes de 3.5 pulgadas, con capacidad de 1 Mb, sin formatear, 720 Kb, formateados en discos de doble cara y 360 Kb para discos de simple cara.
- Su velocidad de transmisión es de 250 Kbits/segundo.
- El tiempo de acceso medio es de 331 ms.
- El tiempo de acceso pista-pista es de 12 ms, y tiene un consumo de energía de 16W.

El aspecto externo, acabado en color negro al igual que el resto de los equipos de esta marca, presenta un aspecto sólido con las siguientes dimensiones: 86 x 270 x 166 mm y pesa aproximádamente 2,5 kg. A pesar de las dimensiones y del peso, esta unidad está diseñada para trabajar en posición vertical, con lo que puede ser colocada en cualquier hueco cerca del ordenador.

El interruptor de puesta en marcha se encuentra en la parte frontal, al igual que un led que indica su funcionamiento y un pulsador de extracción del disco. Lo que más llama la atención es, que se puede ampliar con otra unidad de discos, convirtiéndolos en un periférico sumamente potente, habida

rutinas que sirven para que el *hardware* del *diskette* responda a las órdenes que da el usuario; normalmente es «transparente» al usuario, es decir, este emplea un lenguaje de alto nivel (BASIC) para realizar sus programas o ejecuta programas ya hechos, sin preocuparse de las actividades del Sistema Operativo. Eventualmente usará funciones de éste *(COPY, RENAME...)*, o pasará por él como paso intermedio en la carga de otros programas o lenguajes.

El *Disk BASIC* es una ampliación del MSX BASIC, que nos da acceso a una serie de comandos que podremos usar directamente o incluir en nuestros programas. Su principal ventaja, aparte de la velocidad de acceso al disco, es la posibilidad de trabajar con ficheros aleatorios, o sea, aquellos en que es posible acceder a un determinado elemento sin tener que cargar los demás desde el disco.

En los *diskettes,* al igual que en los cassettes, es posible impedir la grabación a voluntad del usuario. Para ello, los *diskettes* de 5 1/4" disponen de una muesca que se cubre con una etiqueta adhesiva (se proporciona con el *diskette*) para impedir la grabación. En los *diskettes* de 3 1/2" el sistema es más sencillo: hay un orificio que se abre o cierra con una pestaña deslizante; cuando está abierto no es posible escribir.

La zona de memoria donde está el *buffer* para *DMA* puede localizarse viendo el contenido de la variable FILTAB; este buffer tiene 512 bytes de longitud. A continuación aparece la imagen en *RAM* de la *FAT;* para hacer que se cargue basta introducir un comando de *Disk BASIC* que suponga transferencia de información (el más cómodo es *FILES,* que lista los ficheros contenidos en el disco sin alterar programa o variables). La dirección de inicio de la zona *DMA* viene dada por

FILTAB = PEEK(&HF860) + 256 × PEEK(&HF861)

Si se desea examinar el contenido de un sector de un disco determinado se debe usar la instrucción A$ = DSKI$(d,n), donde A$ puede ser cualquier variable de cadena, d será el número del *drive* (0 = drive por defecto o actual, 1 = A, 2 = B...) y n es el número del sector lógico. El contenido del sector quedará en la zona de *DMA.* Esto es muy práctico, ya que así es posible, por ejemplo, estudiar un programa sin cargarlo en memoria. Un caso de especial interés son los programas con extensión *COM,* que se cargan directamente desde el sistema operativo; estos programas empiezan siempre en la dirección &H0100 (con el MSX-DOS se emplean 64K RAM, dejando de actuar la ROM BASIC), por lo que no pueden cargarse directamente desde BASIC.

Lorenzo Hernández

sos contaremos con dos discos de 720 Kb, que conjuntamente suman algo más de ¡1 Mb! Esta operación debe ser sencilla, puesto que existe un hueco expresamente preparado para tal fin.

En la parte posterior se encuentra el cable de alimentación y el conector para el ordenador. La conexión al ordenador se realiza mediante el interface ML-30DC, un controlador que contiene la circuitería necesaria para manejar una ao las dos unidades, así como una ROM con el software que permita un buen funcionamiento de éstas.

El equipo viene soportado por la documentación necesaria para su correcto manejo, funcionamiento y mantenimiento. Esta documentación está repartida en tres manuales, todos ellos en castellano. Dos de estos manuales están dedicados a la unidad de discos y el controlador de la unidad, explicando la utilización, conexión y mantenimiento de este periférico, además se dan unos consejos para el manejo y cuidado de los diskettes.

El último de los tres manuales, el más extenso de todos (111 pág.), está íntegramente dedicado al DISC BASIC. En él, se describe el sistema del disco, los pasos a seguir para comenzar a trabajar con la unidad, la preparación (formateado) de los discos, así como las nuevas instrucciones y funciones añadidas al sistema. El tratamiento de ficheros se comenta a fondo en este manual, desde los conceptos básicos de fichero, hasta los ficheros de acceso directo, incluyendo varios programas de ejemplo para facilitar el aprendizaje y una mejor utilización de las posibilidades del sistema.

En suma, se trata de una alternativa más a la hora de elegir una unidad de disco. Ofrece unas ventajas importantes, como es la de poder incorporar una segunda unidad dentro de la misma carcasa, con lo cual se ahorra un espacio considerable. También cabe destacar el buen diseño del aparato, que permite una buena refrigeración impidiendo que los discos se calienten excesivamente, cosa que suele ocurrir en otras unidades.

SOFTWARE

Titulo: Master of the Lamps
Tipo: Juego
Distribuidor: PROEIN, S.A.
Formato: Cassette

Los juegos siempre tienen que contar con un elemento esencial, la fantasía. Para crear, todos nos servimos de nuestra fantasía e imaginación, y *Master of the Lamps* es un juego que le invadirá en un sueño del que si se descuída no podrá salir.

Un lugar perdido en Persia, escenario de las más majestuosas aventuras, se ha visto invadido por misteriosos enigmas y malignos espíritus que han derrocado de su trono al sultán, y éste ha desaparecido. Pero su hijo, el joven príncipe logró salir a salvo de esa contienda y debemos ayudarle a conseguir lo que por derecho le pertenece.

Había sido un lugar de paz, pero esta banda de espíritus ha sembrado el terror y ha hecho que el reino se traslade a otra dimensión.

Tendremos que buscar nuestro reino por infinidad de laberintos espaciales y pasar duras pruebas de rapidez y atención.

Nuestro príncipe tan sólo contará con una alfombra mágica que le transportará por el espacio, no sabrá qué camino tomar. Los espíritus se han encargado de que sea imposible encontrar el lugar donde han escondido su reino, pero aún así, nos han dado unas pistas y unas pruebas a superar.

El juego consta de tres opciones a elegir, pero añadiríamos que cada una tiene su orden de preferencia. En la pantalla de selección, usted puede elegir entre *Seven Trials, Throne Quest* y *Magic Carpet.*

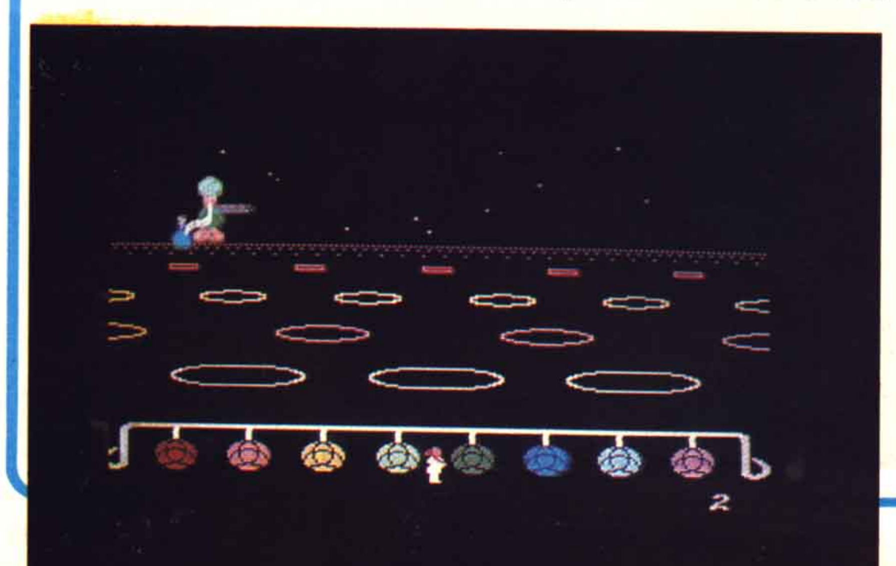
Seven Trials son una serie de estadios o metas que usted debe alcanzar, serán largos e interminables pasadizos de rombos espaciales por los que debe volar lo más rápido posible, pero con mucho cuidado ya que éstos cambian su ruta cuando menos se lo espere. A lo largo de su paseo le acompañará una extraordinaria melodía que al comenzar el juego le animará éste, pero que se convertirá en una pesadilla, ya que debemos tener una concentración muy alta y no despistarnos nunca. Los creadores de este programa, han tenido esta característica en cuenta y han habilitado una tecla al volumen de la sintonía, así esto no será un impedimento para nosotros. Aunque no lo parezca, esta es una de las pruebas a las que se nos somete en el juego, nuestro poder de concentración.

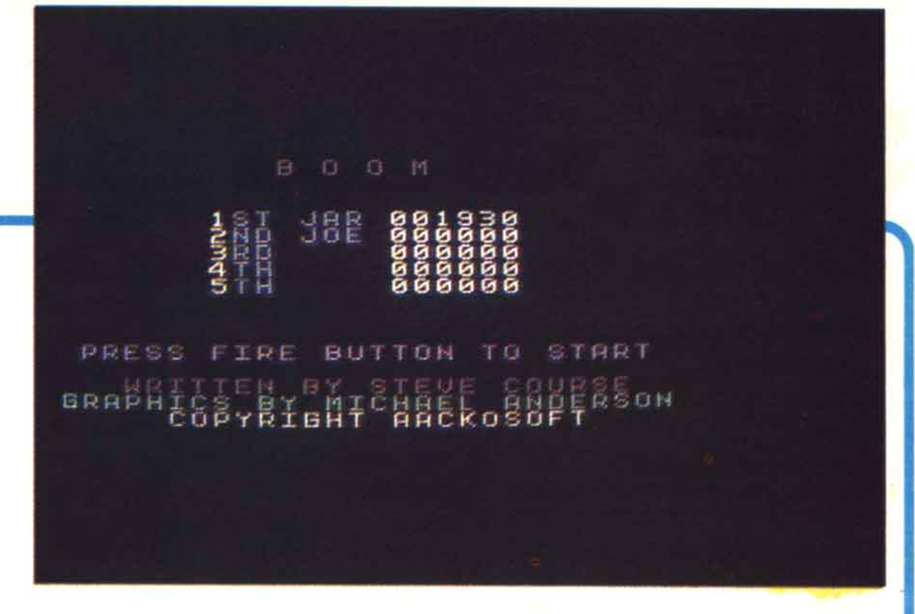
Una vez finalizado el viaje por este túnel de rombos, hecho que no será fácil al menos la primera vez, ya que en el momento que nos choquemos quedaremos destruídos, conseguiremos entrar en la primera prueba, no es difícil pero sí hay que tener muy buen oído y ser muy rápidos. Nos aparecerá en último plano, la visión de nuestro reino invadido por los espíritus que se han adueñado de él, y una fila de lámparas que comenzarán a sonar y a cambiarse de color cuando nosotros deseemos comenzar la prueba.

Esta consiste en golpear el gong, para señalar que estamos listos, una vez hecho esto aparecerá una combinación de sonidos acompañados de la iluminación de las lámparas que contienen esas notas, por lo que debemos fijarnos ya que para superar la primera prueba, tendremos que repetir esta misma combinación antes de que el mago que nos está observando nos haga destruir.

Si logramos superar esta prueba conseguiremos pasar a un estadio superior, el cual nos abrirá las puertas para conseguir nuestro objetivo y nos ayudará a superar las sucesivas pruebas que aún serán más difíciles.

Existe una segunda fase en el juego que es más complicada. *Throne Quest,* o Grado de Maestro, se adquiere una vez que hayamos completado las tres lámparas de siete piezas, no nos vamos a extender en explicaros esto porque si no desvelaríamos el enigma del juego, tan sólo señalaros que el oído y al vista son imprescindibles.





El título de *Master of the Lamps* lo conseguirá cuando haya completado una serie de tres lámparas y consiga destruir el maleficio que cae sobre su reino. Pero aquí no ha terminado el juego, la tercera parte consta de ejercicios de vuelo a fin de conseguir una mayor habilidad al surcar el espacio, es interminable.

Mencionar la singularidad del juego, en cuanto a su sonido, pues como ya sabemos la capacidad de generar sonidos de estos ordenadores, nos permite disfrutar ampliamente de este tipo de juegos que nos ayudan a conocer y despertar más nuestras habilidades, y sobre todo a divertirnos.

Puntuación:
Presentación: 8
Rapidez: 6
Claridad: 6
Adicción: 8

Titulo: BOOM
Tipo: Juego
Distribuidor: Compulogical
Formato: Cassette

¿Qué les vamos a decir de los juegos de marcianitos que ya no sepan? Pues mucho, ya ven.

Los marcianitos, como así se conocen este tipo de juegos, son los pioneros y artífices de los juegos de ordenador y aunque sean miles las versiones que existen de ellos en el mercado, se les sigue considerando como buenos juegos y entre los más divertidos, aún hoy en día.

La sofisticación que han alcanzado algunos de ellos nos demuestran la demanda que existe en el mercado y la adicción que les profesan desde los más pequeños hasta los más mayores.

Si Ud. no tiene aún su juego de marcianitos, ahora, *BOOM!* le brinda la oportunidad de completar su biblioteca de software, ya que este juego es elemento esencial en ella.

BOOM! no es un juego que se diferencie de los demás por las innovaciones que nos presenta, sino más bien se encuentra en la línea clásica y rudimentaria de este tipo de juegos. Pero el entretenimiento es lo importante, y eso desde luego está incluído en este juego.

Tres son las naves con las que cuenta para destuir a la flota enemiga que se le avecina. Numerosos son los comandos y naves galácticas que vienen a por Ud. Debe, principalmente, centrar su atención en que ninguna de ellas le destruya o choque contra su nave, al mismo tiempo que sus reflejos le permiten disparar lo más rápido posible con el fin de acabar con el enemigo.

Conseguir puntos y más puntos será el fin primordial del juego, ya que cada vez que aniquile a una flota enemiga, se sucederán otras y otras y otras..., cada una de ellas con más peligro y mayor velocidad pero nada que no pueda superar.

BOOM! como hacíamos referencia, no es la última novedad en este tipo de juegos, pues el diseño con que se desarrollan los movimientos por ambas partes es muy lenta. Este tipo de programas es ideal para las personas que se inician en los juegos, especialmente para niños, ya que al no ser muy complicado pueden centrar más su atención en el peligro enemigo.

Sin embargo, podemos destacar que es uno de los juegos que ha cuidado el color, ya que podemos diferenciar rápidamente, el ataque enemigo en cualquier momento, y si somos observadores, veremos que normalmente las naves que comienzan su proceso de ataque cambian de color y se proyectan vertiginosamente hacia nosotros, para aniquilarnos.

Mantener la calma en todo momento es esencial, pues no sólo os conducirá a la victoria, sino que os relajará. Se tiene un mal concepto de este tipo de juegos, que sirve para cargar más los nervios y en el noventa por ciento de los casos se utiliza como escape, después de un día duro o un estado de nervios intenso, pero bueno, no es que sea medicinal, tan sólo que sí consigue hacernos pasar un buen rato y divertirnos, se merece tenerlo en cuenta.

Puntuación:
Presentación: 5
Rapidez: 8
Claridad: 5
Adicción: 6

SOFTWARE

Titulo: Invierte y gana
Tipo: Juego
Distribuidor: Dimension New
Formato: Cassette

¿Quién no se ha sentido más de una vez atraído por el mundo de la Economía, las actividades comerciales, las finanzas y el enigmático pero peligroso mundo de la Bolsa?

Golpe de suerte, valores bien invertidos y en el momento adecuado, han creado algunas de las fortunas más famosas e importantes de la economía mundial.

Invierte y gana, no tendrá más repercusiones en un primer momento que la de hacerte pasar un rato agradable, pero a su vez le familiarizará con todos los pormenores que encierra este mundo y enseñará de una forma teórica, las reglas, las bases y los principales campos o mercados de inversión.

ugador: IVAN | Mes: 1 | Año: 1
isponible: 495000 Préstamo: 0
nvertido 0 Pago Mes: 0
Relación cartera de valores

	Sectores	Acciones	Valor total
1	ELECTRICIDAD	0	0
2	BANCOS	0	0
3	SEGUROS	0	0
4	PETROLEOS	0	0
5	INDUSTRIA	0	0
	Total...	0	0

Carnet de Asociado: En vigor
Pulse una tecla para seguir

La propia Bolsa Mundial la podemos considerar como un juego, cuyo fin primordial no es la diversión, sino otras muchas facetas que pueden hacer olvidar a una empresa o elevarla a la cumbre.

Dejando a un lado el aspecto general de la Bolsa, comentaremos las características del programa en sí, tanto técnicas como de contenido.

El objetivo del juego es conseguir unos beneficios líquidos estipulados en un primer momento contando tan sólo con su habilidad, intuición y suerte con las acciones que Ud. va a comprar o vender.

Los participantes pueden ser 1 ó 6, y cada uno de ellos es una persona física o una empresa o persona jurídica, así como a lo largo de éste, también se pueden crear asociaciones dependiendo de la coyuntura económica y el estado del mercado, esto les permitirá a su vez, llevar un segundo control con otros inversores y cambiar el cariz en un momento de la Bolsa MSX.

Todas las personas que vayan a jugar aparecerán como una empresa o entidad, la cual contará con un capital inicial de 1.000.000 de ptas. para invertir en la compra de acciones. A su vez podrán obtener créditos que eleven su capital y puedan hacer frente a las eventualidades que se les presente.

Cinco serán los sectores que formen la Bolsa, divididos a su vez en cinco empresa cada uno: Electricidad, Bancos, Seguros, Petróleos e Industria.

Cada sociedad saldrá al mercado con una venta de acciones no superior a 2.000, a un valor de 100 enteros (500 líquido), las cuales Ud. debe adquirir para conseguir los beneficios estipulados al principio del juego. Recibirá información a lo largo del juego, por el teletipo y por el menú de opciones el cual puede ver en el momento que lo desee, le indicará de una forma aleatoria (&) la situación de los distintos sectores que forman el mercado.

El menú consta de una serie de puntos, siete en total, los cuales utilizará Ud. para indicar las operaciones que desea realizar: cotizaciones, cartera de valores, operaciones bancarias, compra, venta, etc.

Para comenzar el juego, es esencial que haya obtenido el carnet de asociado, carnet con el que podrá efectuar operaciones financieras y por el que tendrá que pagar al inicio del juego 5.000 ptas., renovándolo anualmente.

Si Ud. ha sido un buen inversor, gozará al final de ejercicio anual de unos beneficios, ya que se reparten dividendos. Si por el contrario no ha conseguido superar lo estipulado, habrá perdido.

Otras connotaciones que aparecen en el juego, son la capacidad de hojear en un instante la situación de su empresa, ya que en el menú de opciones y detrás de cada operación le aparecerá una hoja resumen de todas sus operaciones y situación bursátil, así como el período transcurrido.

El teletipo, o nuestro telediario particular, nos proporcionará toda la información necesaria en cuanto a lo que ocurre a nivel mundial que nos pueda afectar directa o indirectamente.

Para aquellas personas que consideren que estos temas carecen de una cierta diversión, podrán comprobar que *Invierte y gana* es todo lo contrario, ya que la simulación es muy buena y el aspecto técnico hace que no sea pesado jugar, pues las numerosas pantallas con las que cuenta, la evolución del programa en sí nos hace sentirnos un gran financiero.

Técnicamente, nos hubiera gustado que el programa fuera más completo, ya que carece de algunos datos de importancia, como el período necesario que debe

transcurrir entre la adquisición y venta de una acción, o la rápida visualización de una empresa que pertenezca a uno u otro sector a la hora de decidirnos en la compra o venta de acciones.

Un efecto a destacar es el juego de colores que aparece a lo largo del programa, nota que hace más agradable este complicado mundo, aunque esto sea un juego.

Invierte y gana simula, en gran medida, el mercado de la Bolsa y sus probabilidades, así como nos enseña a introducirnos en el mundillo de las acciones, ya que hoy por hoy nos vendría bien a todos tener acciones, porque desgravan.

Puntuación:
Presentación: 7
Rapidez: 6
Claridad: 5
Adicción: 6

Titulo: Calculator New
Tipo: Operativo
Distribuidor: Dimension New
Formato: Cassette

El principio de las cosas siempre encierra un pequeño misterio y la Ciencia de la Matemática también lo tiene, ¿quién les iba a decir a los egipcios que un día una máquina les iba a quitar sus quebraderos de cabeza?

Los números, las fórmulas, operaciones aritmeticológicas, ya no nos darán más dolores de cabeza, ya que nuestro ordenador y *Calculator New* forman el mejor equipo a la hora de sacarle de un apuro, como son esos pequeños detalles de las operaciones.

Nuestra calculadora de repente ha crecido, pero no como una calculadora sin mas, sino como una calculadora científica.

Calculator New ha sido pensado para todas aquellas operaciones que Ud. necesitará realizar en su trabajo profesional, o como hobby.

El tiempo, hoy en día, es indispensable para Técnicos, estudiantes, programadores, que requieren de una herramienta sofisticada, pero a la vez rápida y sencilla de manejar.

Es cierto que una de las facetas de un ordenador es la de ayudar o facilitar la labor, es por ello que con este programa podrá realizar dos tipos de operaciones que son:

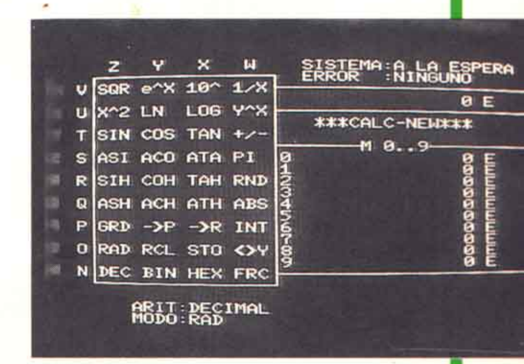
– Operaciones Simples: Suma, resta, multiplicación, exponenciación, raíces, logaritmos, etc.

– Operaciones Especiales: Que engloban las anteriormente citadas, junto con los sistemas de números (hexadecimal, binario, etc.), y operaciones angulares (grados o radianes).

La capacidad del programa es mayor que la de una calculadora normal, ya que cuenta con una memoria de hasta diez cifras y 36 funciones especiales. Su presentación es clara y sencilla para una persona que esté acostumbrada a realizar este tipo de operaciones, aunque es conveniente leer cuidadosamente el manual de instrucciones para conseguir el resultado que deseamos.

Las operaciones se realizarán en una hoja a modo de fichero, con diez apartados en caso de querer utilizar la memoria. A su vez obtendremos una visión total de las operaciones que deseemos realizar, a excepción del sistema numérico en que queramos operar, esto no quiere decir que sea restrictiva para operar en un mismo sistema, sino que se estipula antes de realizar una operación, de lo contrario se regirá por el sistema decimal.

¿Cómo elegir las operaciones? En la misma hoja que obtendremos los resultados, nos aparecerá un cuadro, en donde se encuentran especificadas las operaciones, divididas en filas y columnas



a modo de cuadro cartesiano, bastará con dar las coordenadas y los dígitos con que queramos operar y obtendremos los resultados.

Existen unas operaciones más complejas que otras, es por ello que mientras la máquina realiza la operación, aparezca la palabra «Procesando». Otras notas a tener en cuenta, es que Ud. puede hacer en cualquier momento una reinicialización de la calculadora, por que haya ocupado su memoria, etc. Asimismo, el programa *Calculator New* cuenta con un analizador sintáctico de la calculadora, que en caso de error por su parte no realizará la operación, al haber pulsado una tecla equivocada.

Un programa, además de ser útil, debe ser diferente y cada nuevo programa ha de tener una característica que lo diferencie de los demás. *Calculator New* es también un reloj y calendario, que le informará cuando Ud. lo necesite, por si se queda ensimismado o el

el mejor softwa





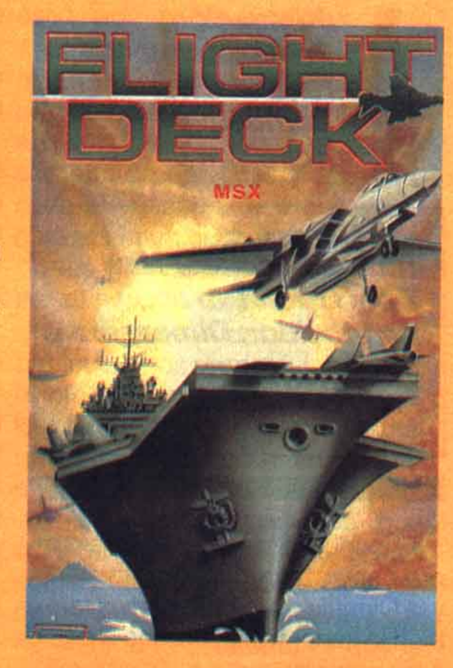
DROME

Entretanto en DROME, un Super-ordenador, debes encontrar y eliminar los sofisticados sistemas de defensa y supervivencia.

Has de elegir uno de los cuatro sectores que constituyen los mecanismos de defensa de esta terrorífica máquina.

Un atractivo juego de acción, donde se pone a prueba la capacidad de la máquina y del jugador.

Precio de venta 2.600 ptas. (IVA incluido)



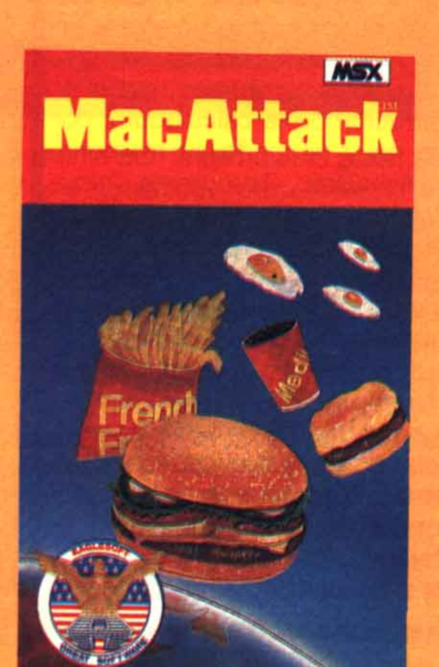
FLIGHT DECK

Sienta la emoción del golfo de Sidra en casa.

FLIGHT DECK es un juego de estrategia y habilidiad en el que tendrás que desmantelar las bases enemigas.

Al mando de un portaaviones donde dispones de 10 unidades de combate... y poco tiempo.

Precio de venta 2.600 ptas. (IVA incluido)





MC-ATTACK

Ayuda a Fredy, el Rey de la Hamburguesa a preparar el suculento manjar que hace las delicias de los comensales.

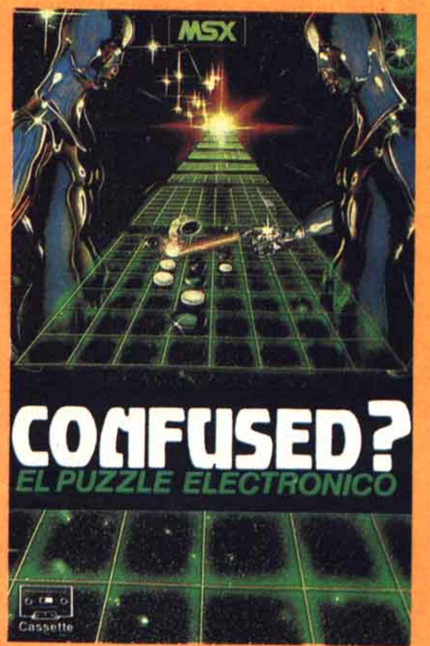
Ten cuidado con las salchichas grasientas y los huevos escurridizos que intentarán arruinar tu exquisito plato.

Defínete con la pimienta y procura hacer el mejor número de hamburguesas posible.

... Buen provecho.

Precio de venta 1.900 ptas. (IVA incluido)

re para MSX



CONFUSED?

Es el puzzle electrónico.

El objeto del juego es resolver 10 puzzles con distinto número de piezas, a elegir, pero todas... MOVIENDOSE.

Pon a prueba tu inteligencia y capacidad de deducción para solucionar algunos de estos entretenidos rompecabezas.

Precio de venta 2.600 ptas. (**IVA incluido**)

NORTH SEA HELICOPTER

Una explosión en una plataforma en el mar del Norte arroja a los hombres a un destino incierto...

Empieza una carrera contrareloj para salvarles de su fatal situación.

Tienes que convertirte en un piloto experimentado para mantener el control del helicóptero... El tiempo empeora.

¿Crees que cumplirás la misión?

Precio de venta 2.600 ptas. (**IVA incluido**)









SPACE RESCUE

Desciende de tu nave nodriza a la superficie lunar e intenta rescatar a los hombres extraviados.

Ojo con los meteoritos que te destruirán cuando intentes regresar.

Disponer de nuevas plataformas para culminar con éxito la misión.

Desgraciadamente tu nave nodriza está bloqueada por unidades de combate enemigas... Intenta destruirlas.

Precio de venta 2.600 ptas. (**IVA incluido**)

Solicita tu programa favorito en tu tienda especializada habitual o recorta y envía este cupón a INFODIS, S. A., c/Bravo Murillo, 377. 5.° A - 28020 MADRID.

NOMBRE ____________________

DIRECCION ____________________

POBLACION __________ PROVINCIA __________ C. P. __________

Forma de pago: CHEQUE ☐ CONTRAREEMBOLSO ☐

TARJETA DE CREDITO: American Express ☐ VISA ☐ INTERBANK ☐

Número de mi tarjeta ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

CADUCIDAD ____________________

Firma ____________________

TITULO	CANTIDAD	TITULO	CANTIDAD
DROME FLIGH DECK MC-ATTACK		CONFUSED? NORTH SEA HELICOPTER SPACE RESCUE	

infodis, s.a.

SOFTWARE

tiempo es una de sus grandes preocupaciones.

Todos los programas que faciliten una labor, no pueden ser criticados, pero sí diremos algunos puntos que se pueden solucionar para generaciones venideras.

La monotonía del color en la pantalla, puede llegar a cansar la vista de una persona que dedique varias horas ante su monitor, ya que si vamos a ocupar la memoria de nuestro ordenador, convirtiéndolo en una calculadora, no va a ser para diez o quince minutos, es por ello, que se podía incluir una opción del color de fondo a elegir, no porque uno u otro sea mejor, sino para distraer un poco la mente matemática.

Calculator New es un programa muy útil, sobre todo si su calculadora no tiene la capacidad que desea. Gracias a la visualización de las operaciones en memoria, puede sacar una conclusión rápida del problema o de la situación matemática.

Puntuación:
Presentación: 6
Rapidez: 7
Claridad: 7
Adicción: –

Titulo: Hopper
Tipo: Juego
Distribuidor: Compulogical
Formato: Cassette

Hopper o el clásico de lujo. Hopper es una simpática rana. Sí, como suena, una rana muy especial, es inteligente.

¿Qué le parece convertirse por unos momentos en un anfibio?

Es este mundo hay que probarlo.

Hopper es un juego sencillo pero a la vez diferente.

Numerosos son los peligros que se le presentarán y mucha la habilidad que deberá demostrar para conseguir finalizar el juego.

El instinto de los animales no falla a la hora de conseguir alimentos, pero una ranita perdida en medio de una autopista, verdaderamente le va a ser un poco difícil.

Hopper, nuestra rana, sólo come moscas y se ha empeñado en comer un tipo de mosca que se encuentra al otro lado de la autopista, en la ribera de un río cercano.

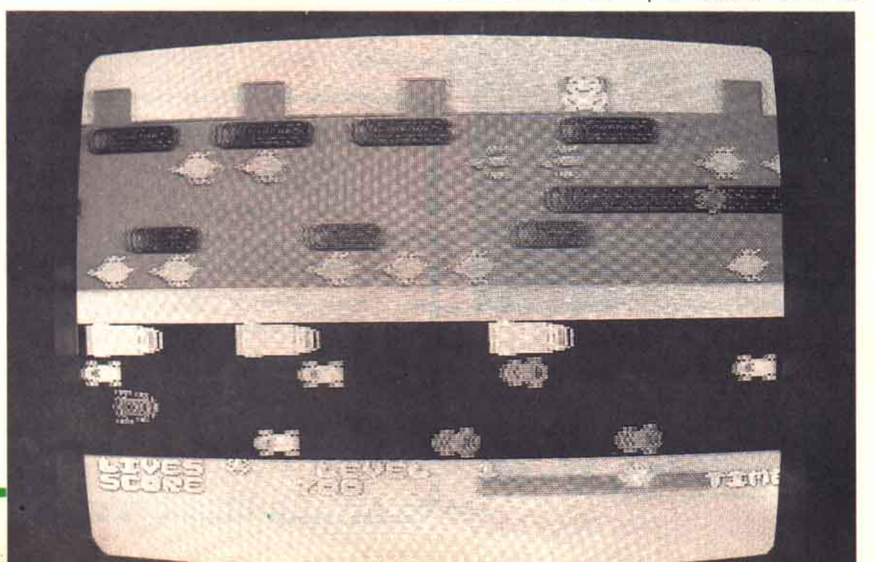
El juego comienza en el momento en que Ud. coloca un anca en la autopista, hecho que debe pensar con tranquilidad, porque la velocidad de los coches no es pequeña. Para los más pequeños, a quienes va dirigido este tipo de juegos, entra un factor muy importante, la rapidez de reflejos. Con esto queremos decir que, en ocasiones se podrá estar delante de un coche siempre y cuando sepamos en qué momento se puede saltar. En esto hay que ser rápido y decidido, pues cualquier titubeo puede costarnos la vida.

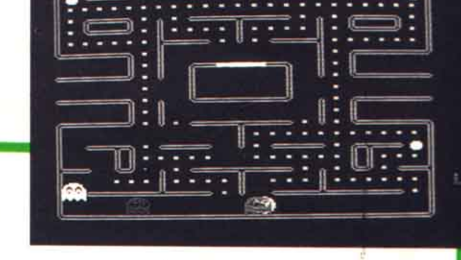
Una vez superada esta prueba, os encontraréis a orillas de un río y al otro lado, se encuentra la comida, una sabrosa mosca que irá revoloteando de un lugar para otro, porque nuestra ranita no está hambrienta y no tiene fuerzas para nadar, así que habrá que ayudarla a cruzar por encima de diversos objetos que lleva la corriente.

Comienza la aventura. Siempre que pisemos un tronco, correremos el grave peligro que se hunda y que nos arrastre al fondo del río, perdiendo así una vida. Pero es difícil que ocurra porque tenemos tiempo suficiente para saltar del tronco hacia alguna zona más segura o volver atrás. Con las tortugas debemos tener más cuidado, ya que ellas pueden nadar y a veces hacen lo posible para hacerlo bajo el agua, de esta manera si estamos en una de ellas, deberemos saltar cuando éstas empiecen a sumergirse, de lo contrario pereceremos.

No es difícil capturar la mosca, sin embargo, el secreto se encuentra en mantenerse el mayor tiempo posible en el río y no dejar que nos ahoguemos.

Contaremos con tres vidas al inicio del juego y cada vez que nos comamos una mosca contaremos con un bonus que aumentará la





puntuación. Si conseguimos finalizar una etapa, pasaremos a la siguiente, más difícil pero más divertida.

De cualquier manera, el juego es muy entretenido y sencillo de acabar, aunque las características gráficas dejan algo que desear, ya que nuestra ranita (de color verde) se confunde con el río, cuyo tono verdáceo no tiene mucha semejanza con el de un auténtico río.

La agrupación de tantos elementos en una sola pantalla, perjudicará sensiblemente a los más pequeños, pero nada que no se puede corregir con el tiempo. *Hopper* es un buen juego.

Puntuación:
Presentación: 7
Rapidez: 7
Claridad: 8
Adicción: 8

Titulo: Oh Shit!
Tipo: Juego
Distribuidor: Compulogical
Formato: Cassette

¿Qué es un comecocos? El significado de esta palabra es, hoy en día, tan amplio que no sabríamos a qué referirnos con ella. Pero sí existe un origen, o por lo menos un comienzo, y está en aquellas maquinitas de cinco duros que, literalmente nos comían el coco.

Podemos decir de este juego que ahora comentamos, fue el pionero de los juegos, o por lo menos el primero que conocimos casi todos, cuando el mundo de los ordenadores empezó a inundar nuestros hogares.

Las características del juego son de todos conocidas, pero si hay alguien que comience a introducirse en este mundo de locos de los juegos de ordenador, indicaremos las directrices principales y de qué se trata.

En la pantalla se crea un laberinto delimitado por barreras de colores que forman pasadizos y por los que debemos movernos a fin de conseguir escapar primero y segundo, obtener la mayor puntuación.

La visión del juego es la misma que ofrecen las máquinas de los bares. Estos se persiguen unos a otros y lo único que logran es crear una confusión y ponernos más nerviosos. Pero nuestra bolita comedora no para y cada vez quiere comer más. Por este motivo, no debemos detenernos, ni dejarnos comer nunca, de lo contrario se acabaría la partida.

¿Qué es lo que vamos a comer? Unas veces serán cerezas, otras fresas, plátanos, depende del nivel en el que estemos.

En cada nivel, éstas se irán haciendo más escasas y nuestros perseguidores más rápidos, dificultando la tarea de entrar en la despensa.

Existirán situaciones en que comamos más rápido de lo normal y la máquina nos compensará con bonus o permitiéndonos pasar por encima de los fantasmas sin que estos nos coman, momento en que cambiará la pantalla de color para indicarnos que debemos aprovechar ese tiempo para conseguir más puntos.

Esto es el juego en esencia, no tiene más elementos, pero tampoco le faltan. Es por ello que, aunque hoy tengamos en el mercado juegos más sofisticados, recurramos a este como uno de los más divertidos y menos dificultosos, que al fin y al cabo de eso se trata, porque nadie toma un juego como un deber.

Las características técnicas del programa son muy primarias, eso sí, en esta versión se han cuidado a fondo aquellas pequeñas deficiencias que los primeros juegos podían presentar, como la falta de sonido, color, rapidez, etc.

Qué familiar resulta el sonido que hace nuestro muñequito al engullir esas frutas o esferas que aparecen por la pantalla, porque no sólo devora fruta, sino todo lo que se ponga en su camino. Olvidar este sonido característico le quitaría al juego todo su encanto, es por ello que estas nuevas creaciones conserven los convencionalismos propios de este juego, ya que de lo contrario se convertiría en una desvirtualización más o un simple juego para pasar el rato.

Los gráficos que constituyen las pantallas han mejorado en cuanto a realización y se han perfeccionado los detalles de los *sprites* así como el movimiento, siendo aún más uniforme.

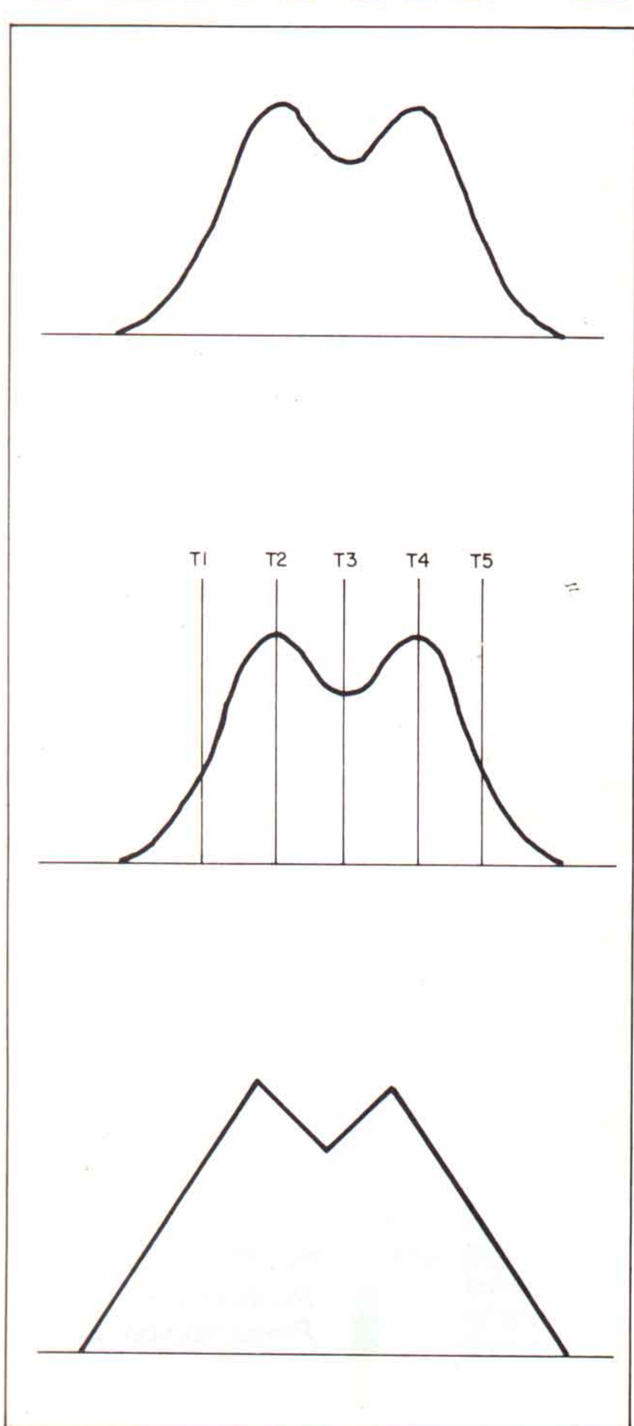
Recomendamos la utilización de joystick, debido a que se requiere una mayor sensibilidad en los movimientos.

Siempre que se dice volver atrás, se piensa en retroceder, aquí no ocurre esto, sino que se ha avanzado ofreciéndonos al alcance de nuestras manos el juego más sencillo pero a la vez más entretenido, el comecocos.

Oh Shit! es, por esta razón, un juego único que no debemos olvidar ni dejar pasar.

Puntuación:
Presentación: 8
Rapidez: 7
Claridad: 8
Adicción: 9

Síntesis de voz para l



(II)

os MSX



En el capítulo anterior pudimos generar un sonido mediante el uso del *port* del altavoz. Ahora nos toca saber cómo poder crear cualquier tipo de onda usando dicho *port*.

En la figura 1 tenemos una forma de onda que muy bien podría corresponder a una vocal cualquiera. Basándonos en la increíble rapidez del código máquina podemos hacer que dicha onda, o al menos una muy parecida a ella, sea generada por el altavoz del televisor. Para entender de qué forma se puede hacer esto, dividamos (fig. 2) dicha onda en 5 instantes de tiempo a los que llamaremos t1, t2, t3, t4 y t5. Antes de generar la onda, la membrana del altavoz se encontrará relajada por lo que para llegar a t1 sólo hay que hacer contraer dicha membrana. Como el tiempo transcurrido entre el inicio y t1 es muy corto, al altavoz no la ha dado tiempo de contraerse totalmente, de forma que para llegar a t2 tenemos que volver a contraer la membrana (o simplemente dejar que se contraiga del todo). En t3 el nivel de amplitud de sonido es menor que en t2 por lo que lógicamente ahora nos toca relajar el altavoz. En t4 hay que volver a contraerlo y en t5 dejaremos que la membrana del altavoz se relaje totalmente.

Si fuésemos capaces de ver la forma de onda generada de esta manera por el ordenador, la imagen se parecería bastante a la de la figura 3.

Vamos ahora a lo más importante; codifiquemos dicha onda. Sea el valor «1» el que nos indique que hay que contraer la membrana del altavoz y «0» que hay que relajarla. De esta forma podremos codificar la onda de la figura 1 en un valor de 5 *bits* los cuales serían:

1 1 0 1 0 0

Ahora nos hace falta conocer el sistema que nos codifique a binario cualquier tipo de onda. Lógicamente interminable y aburridísimo el que nosotros tengamos que codificar «a mano» todos los sonidos que nos interesan por lo que vamos a usar un sistema, ya disponible por el ordenador, que nos hará la tarea mucho más fácil.

El port de cassette

En los ordenadores MSX el *bit* 7 (hay que tener en cuenta que en un octeto el primer *bit* es siempre el número cero) del *port A2h* es el encargado de tomar los valores correspondientes, «1» ó «0», según el sonido entrado por la clavija del cassette. Este *bit* tomará valor «1» si el sondio entrado tiene mayor intensidad (o es saturante) que el inmediatamente anterior y será «0» si su intensidad es menor. Funciona, pues, de la forma más útil para nosotros ya que realiza lo mismo que el *port* del altavoz pero de forma contraria.

Realicemos un programa en código máquina que nos demostrará la veracidad de lo anteriormente expuesto. Para los que no gusten del código máquina les diremos que este programa «pasará» directamente y uno a uno los datos entrados por el *port* de cassette al *port* de altavoz de forma que el ordenador trabajará como

un «repetidor» de todo cuanto «escuche» a través de la clavija del cassette. El programa *BASIC* que viene a continuación del máquina será el encargado de cargar el programa máquina en memoria. Usar pues este programa para hacerlo funcionar.

```
      ORG 55000 ; COMIENZO
      DI
A1:   LD E,0
      IN A,(=A2)
      RLA
      RL E
      LD A,E
      CALL =135
      JP A1
```

Expliquemos un poco el funcionamiento del programa. Al principio inhibimos las interrupciones y borramos el registro E. Cargamos en el acumulador el valor entrado por el *port* de cassette y rotamos al acumulador para que el valor del *bit* 7 pase a la bandera de acarreo. Al rotar ahora el registro E (RL E) el valor del acarreo pasará a dicho registro lo que significa que al final de la operación, el registro E tiene el mismo valor que el *bit* 7 del *port* de cassette. Hay otra cosa que también es nueva: en vez de usar la pareja de instrucciones *OUT* al *port ABH* ya conocidas hacemos una llamada a la rutina *135h* de la *ROM MSX* (más concretamente pertenece al *BIOS* de la *ROM*) la cual contraerá la membrana del altavoz si el registro A tiene valor 1h o la relajará si es cero. De esta forma nos ahorramos unas cuantas instrucciones.

El programa *BASIC* que carga el máquina anterior es el siguiente:

```
10 CLEAR 256,54999:DI=55000
20 FORI=0 TO 14
30 READ J:POKE (DI+I),J
40 NEXT I
50 DATA 243,30,0,219,23,203,
19,123,205,53,1,195,217,214
```

Una vez cargado el programa máquina podemos guardarlo en cassette o disco mediante *BSAVE "SONI2",55000,55014* y cargarlo desde el mismo medio mediante *BLOAD"SONI2"*.

Para ejecutar el programa sólo tendremos que hacer:

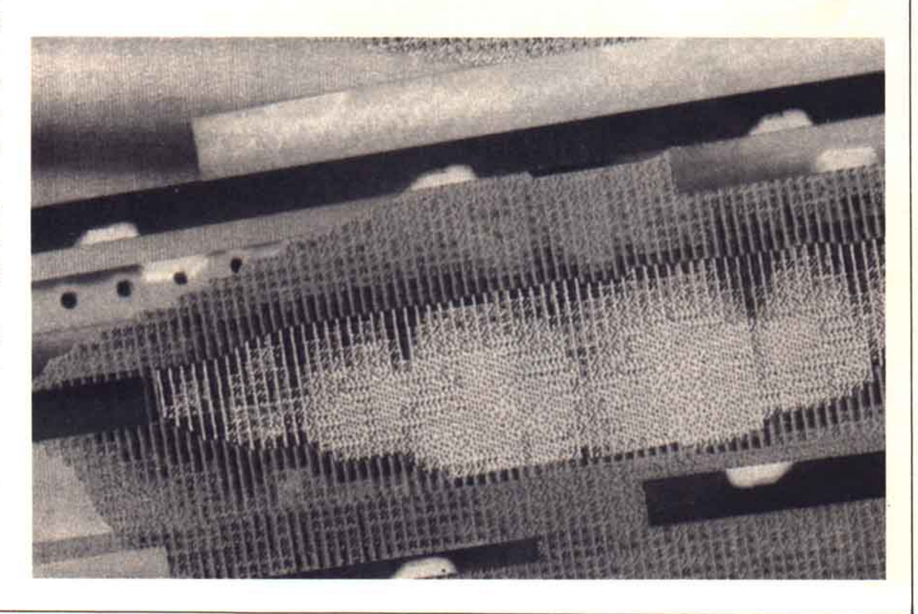
```
defusr=55000
x=usr(l)
```

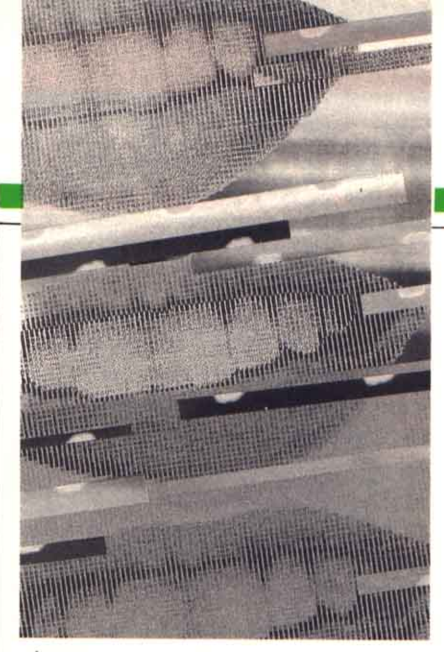
Una vez ejecutada esta pareja de instrucciones, el cursor desaparecerá y no escucharemos absolutamente nada. Para que el ordenador repita lo que oíga, deberá oír algo. Para ello lo mejor es coger una radio normal y sintonizar una emisora. Una vez hecho esto introducir la clavija usada para la carga de programas desde cassette en el conector *"EAR"* de la radio. Finalmente dar volumen a la radio y al televisor. si todo se ha hecho de forma correcta escucharemos la radio a través del televisor y no es la radio la que habla, sino el ordenador, ya que es éste el que se encarga de repetir lo que escucha.

Un programa sintetizador de frases: VOCALS

Llegamos a la parte final y seguramente la más ansiada por todos. Vamos a exponer y explicar el programa *VOCALS* el cual, como ya se adelantó, va a ser capaz de decir frases habladas con bastante claridad. *VOCALS,* no obstante, no quiere decir más que un programa ilustrativo sobre cómo se puede digitalizar la voz en un MSX. Es, por tanto, el lector el que debe ampliar a *VOCALS* ya que éste ha sido diseñado para que funcione incluso en los ordenadores de menor capacidad.

Vamos a dividir el programa anterior en dos partes que son fundamentales: la primera la llamaremos «receptora» y es la encargada de convertir a *bits* el sonido entrado. La segunda se denominará «transmisora» encargada, como ya sabes, de convertir dichos *bits* a sonido. Ahora vamos a crear una tercera parte, que llamaremos «memorizadora», al cual introduci-





rá en memoria los *bits* codificados por la receptora.

El programa que exponemos a continuación funcionará de la siguiente manera:

1. Mediante un sistema sencillo que en breve explicaremos, el ordenador captará del usuario una frase de un segundo de duración, suficiente como para decir «HOLA» o «BUENOS DIAS» por ejemplo.
2. Al mismo tiempo, la parte receptora codificará a *bits* dicha frase y la memorizadora unirá 8 *bits* para formar un octeto e introducirlo en la memoria.
3. Para finalizar esta primer parte se retornará al BASIC.
4. Cada vez que queramos y mediante una instrucción del tipo *USR(l)* ejecutaremos sólo la parte trasmisora, la cual recuperará los octetos de la memoria, los desglosará en *bits* y los codificará a sonido. Fácil, ¿no?

Veamos el listado ensamablador del programa y expliquémoslo un poco. Este programa sólo será capaz de memorizar una frase de algo más de un segundo de duración, pero debido a la poca memoria que necesita el programa, éste podrá ser introducido y ejecutado en ordenadores MSX de 16Kb. Los lectores avispados serán capaces de ampliar el programa de forma que acepte un mayor número de frases.

LISTADO ENSAMBLADOR DE VOCALS

```
       ORG =C5000 ; ORIGEN
          PARTE RECEPTORA
E1:    CALL =9F
       AND A
       JP Z, E1 ; ESPERA PUL-
          SAR TECLA
       DI ; TECLA YA PULSADA
       LD HL,BUFF
       LD BC,1024 ; NO. OCTE-
          TOS A RECOJER
E4:    LD D,8 ; CONTADOR BITS
       LD E,0
E3:    IN A,(=A2) ; ESCUCHA
          CASSETTE
       RLA
       RL E
       EXX
       LD B,30
E2:    DJNZ E2 ; PEQUEÑA ES-
          PERA BITS-BITS
       EXX
       DEC D
       JP NZ, E3 ; COJIDO OC-
          TETO COMPLETO?
       LD A,E ; SI, GUARDAR
       LD (HL),A
       INC HL
       DEC BC
       LD A,B
       OR C
       JP NZ? E4 ; RECOJIDO
          1024 OCTETOS?
       EI ; SI
       RET ; RETORNA BASIC
       DI ; ORIGEN PARTE
          TRANSMISORA
          ; DIRECCION REAL
          C52Ch
       LD HL,BUFF
       LD BC,1024
TX3:   LD D,8
       LD E,(HL) ; COJE OCTETO
TX1:   EXX
       LD B,30
TX2:   DJNZ TX2
       EXX
       XOR
       RL E
       RLA
       CALL =135
       DEC D
       JP NZ,TX1
       INC HL
       DEC BC
       LD A,B
       OR C
       JP NA,TX3
       EI
       RET
BUFF:  NOP ; COMIENZO TABLA
          DE DATOS
```

El programa es muy parecido al tratado anteriormente. Al principio se espera a que se pulse una tecla usando la rutina del *BIOS MSX 9Fh* para empezar a recojer datos. Estos se irán uniendo uno a uno hasta formar un octeto, el cual se introducirá en memoria en una tabla cuyo comienzo es *BUFF*. Una vez introducidos 1024 octetos, el programa retorna al *BASIC*. La parte transmisora funciona igual que la receptora, sólo que aquí se hace una llamada a la ya conocida dirección 135h y no se introducen datos en memoria, sino que se sacan, como era de suponer.

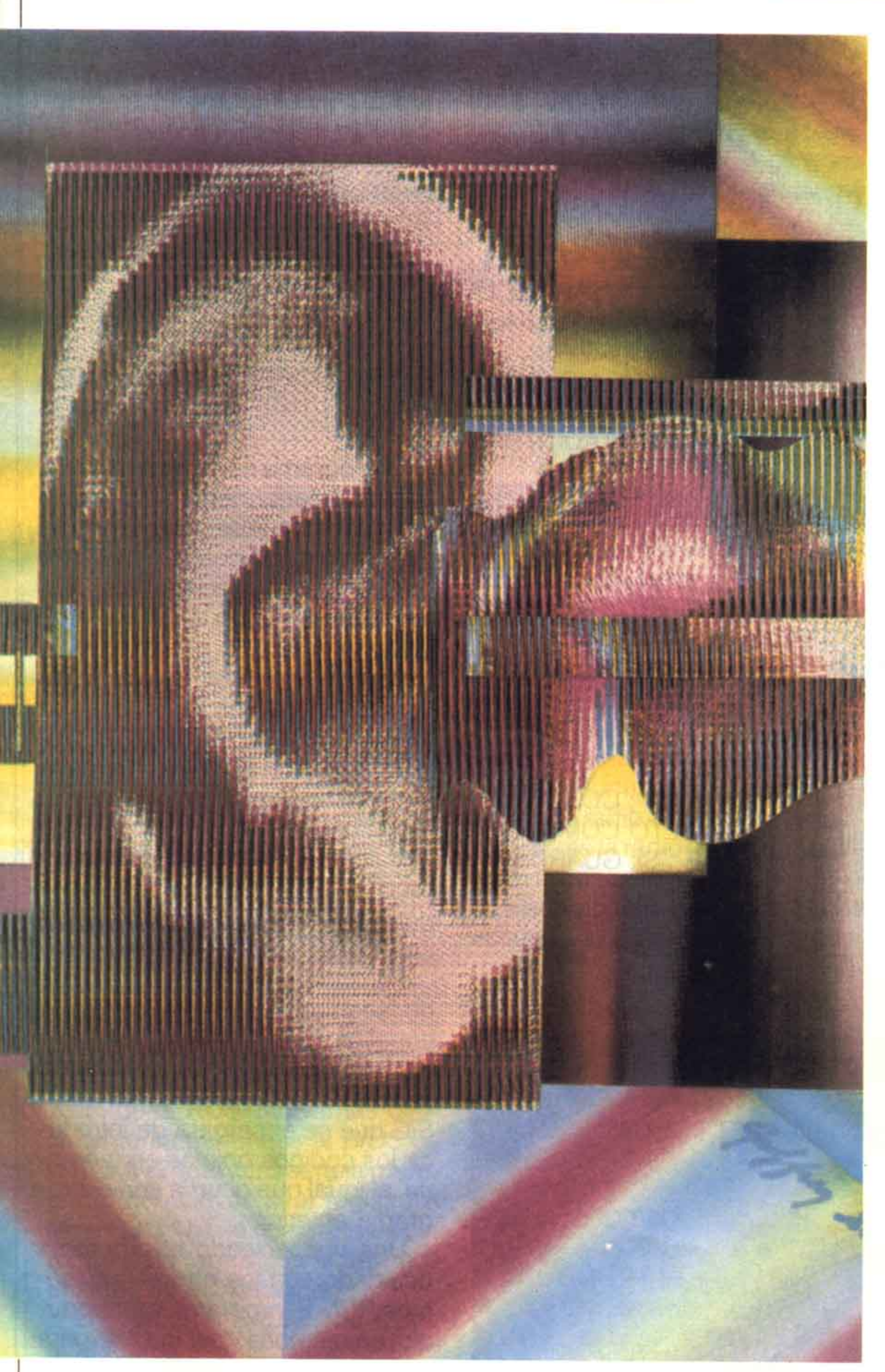
Pensando en los lectores que no sean amantes del código o aquellos que aunque siéndolo no dispongan de un ensamblador, hemos realizado un programa *BASIC* que se encargará de introducir los códigos objetos en memoria, al igual que ocurría con el programa anterior.

Una vez tecleado el programa (¡cuidado con los errores!) pulsamos *RUN*. El programa se introducirá en memoria y estaría listo para ser guardado en cassette (o disco) mediante *BSAVE"VOCALS",&HC500,&HC550.*

El programa máquina se podrá recuperar mediante *BLOAD"VOCALS".*

Veamos cómo usarlo. Hay que

utilidades



colocar un sistema amplificador con micrófono a la entrada del cassette del ordenador. Como esto no va a ser siempre posible se puede seguir otro método que también da buenos resultados (seguir esta orden minuciosamente).

1. Coger una grabadora de cassette que disponga de micrófono e introducir en ella una cinta en la cual podramos grabar algo, o sea, que no esté protegida contra grabación.

2. Introducir la clavija que proviene del ordenador y que usamos para cargar programas desde cassette (generalmente de color blanco) en el conector «*EAR*» o «*EARPHONE*» de la grabadora.

3. Pulsar la tecla *PAUSE* de la grabadora. Esto no es imprescindible, ya que lo único que evitaríamos es que la cinta introducida se grabase con la frase que diremos delante del micrófono. La cinta introducida sólo sirve para poder pulsar la tecla *REC* y *PLAY* de la grabadora, cosa que no sería posible si no existiese en su interior una cinta.

Ya está lista la parte que nos amplificará la voz. Ahora sólo nos queda preparar al ordenador. Para ello tecleamos:

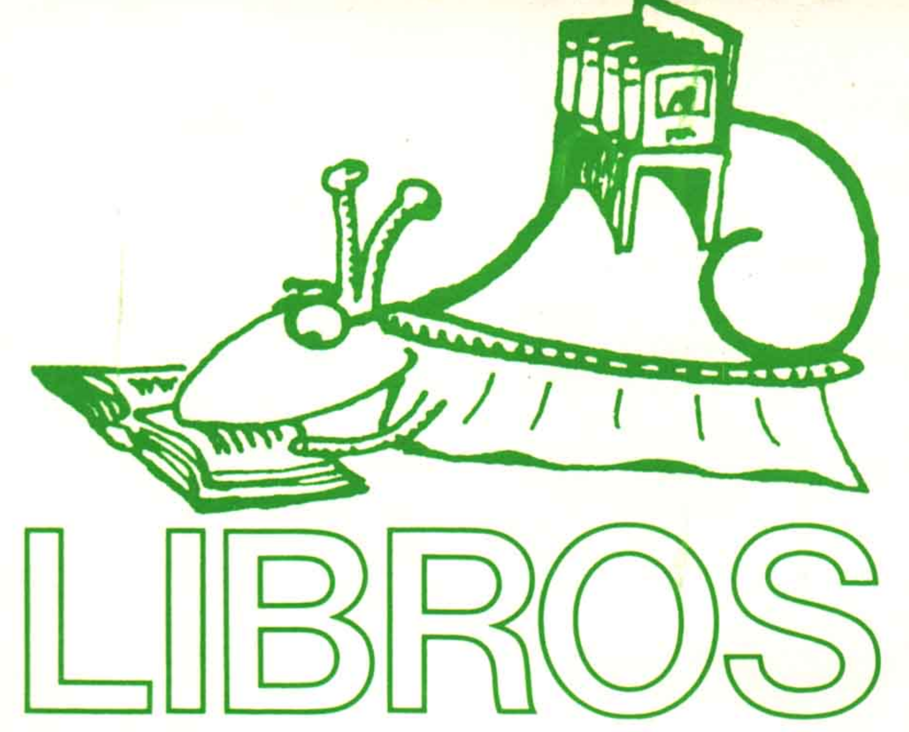
```
DEFUSR=&HC500
X=USR(I)
```

El ordenador estará esperando a que nosotros le pulsemos cualquier tecla para empezar a codificar la frase. Apretamos las teclas *REC* y *PLAY* de la grabadora y después (siempre después) cualquier tecla del ordenador. Inmediatamente pulsada esta última tecla decimos la frase. Una vez terminada la recepción aparecerá el archiconocido "OK" en la pantalla. La frase ha sido codificada y memorizada y podemos escucharla cada vez que queramos, para ello ajustar el volumen del televisor y ejecutar:

```
DEFUSR=&HC52C
X=USR(I)
```

La experiencia le enseñará que mientras más vocalice más perfecta será la reproducción de la frase. Todavía podemos experimentar mucho más...

Juan Jiménez León



LIBROS

Título: Computación Interactiva: Sistemas. Diálogos
Autor: Richard A. Watss
Editorial: Paraninfo
Páginas: 95

En el complicado mundo de la programación y el diseño de aplicaciones, es necesario abarcar una amplia serie de conocimientos, desde diversos lenguajes para desarrollar el software hasta conocimientos de hardware para utilizar mejor los recursos disponibles. Pero hay un campo que tradicionalmente ha estado bastante descuidado, aunque cada día se le presta mayor atención, y éste es el diseño del sistema interactivo, es decir, la comunicación entre el conjunto ordenador-programa y el operador o usuario ocasional.

Este libro es el resultado de una serie de estudios en el campo del Proceso controlado por el usuario, en los que se ha recurrido a las prácticas más modernas y a las últimas ideas de los investigadores. Uno de sus objetivos principales es ayudar a los diseñadores de sistemas allí donde haya ordenadores utilizados por personas inexpertas. Al mismo tiempo, el libro servirá a los diseñadores de sistemas dedicados a usuarios más regulares y a los directores de esos diseñadores, para que puedan justificar las inversiones que se hagan en el desarrollo de los sistemas que se adaptan al usuario.

En realidad no se trata de un libro sobre programación, sino de un compendio de consideraciones que cualquiera que desee diseñar un programa debería hacerse incluso antes de pensar en el programa en sí. Por tanto, se encuadraría mejor en el área del análisis.

Tras una introducción general al tema del desarrollo de sistemas interactivos, se aborda cada uno de los aspectos de la comunicación entre el ordenador y el usuario, contemplando una serie de posibilidades en función de los recursos disponibles, como pueden ser el uso de menús, un sistema de pregunta respuesta, edición de los datos presentados en pantalla.



En resumen, se trata de un libro pequeño en tamaño pero denso en ideas, muy interesante para todo aquél que se preocupe de que sus programas estén bien acabados y sean sencillos de usar. La traducción es buena, lo que ayuda a digerir sin grandes problemas los conceptos expuestos.

Título: Inteligencia Artificial y Sistemas Expertos
Colección: Biblioteca Básica Informática
Editorial: Ingelek, S.A.

Desde el inicio de la era de los ordenadores, los especialistas en informática han tratado de desarrollar técnicas que permitan a las computadoras actuar como lo hace el ser humano. Una de las bases de apoyo de esta nueva forma de diseñar un programa es la Inteligencia Artificial.

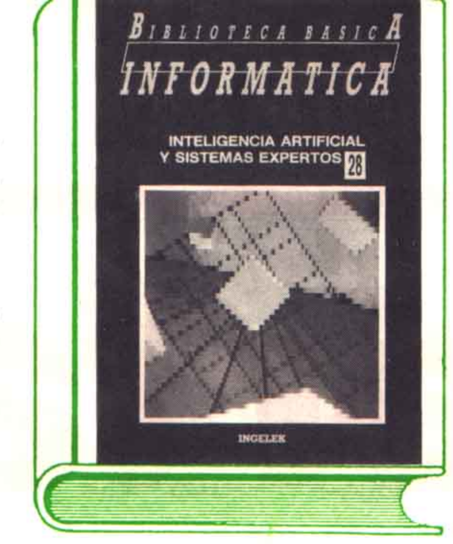
Desde luego, el intento de construir máquinas que presenten las capacidades que asociamos con la inteligencia humana no es nada reciente: ya en el siglo XVIII se construyeron los primeros autómatas que intentaban emular aspectos parciales. La causa del relanzamiento de la Inteligencia Artificial provino de la falta de algoritmos matemáticos sencillas, tales como el reconocimiento del lenguaje, los problemas de diagnosis (enfermedades, geológicos) o el reconocimiento visual de objetos.

El término Inteligencia Artificial se presta en cierta medida a interpretaciones erróneas y, tal y como ocurrió con los «Cerebros Electró-

nicos», quizá dentro de unos pocos años esta denominación haya desaparecido y surjan otros términos más acordes con la realidad de las investigaciones.

Hoy en día los programas más avanzados son más sistemas basados en el conocimiento de expertos que sistemas inteligentes, ya que sus capacidades son, aunque de una cierta potencia, muy limitados en sus aplicaciones, restringiéndose a temas concretos. Tales com los ya muy famosos sistemas de diagnosis de enfermedades infecciosas o de propecciones geológicas, que están siendo aplicados con mucho éxito por numerosos hospitales o por las compañías petrolíferas.

Por todo ello, el tema de la Inteligencia Artificial se halla revestido de gran importancia y resulta ser de candente actualidad. Para que usted, lector, no se quede atrás en



el progreso, este libro le ayudará a comprender el desarrollo de la Inteligencia Artificial, los principios sobre los que se basa, las estructuras de almacenamiento y tratamiento de la información utilizadas.

De hecho, tras una breve introducción a la Inteligencia Artificial, se introducen en él una serie de conceptos básicos fundamentales para comprender y asimilar la información contenida en los capítulos siguientes: Representación del conocimiento, Estudio del motor de inferencia, Procesadores de Lenguaje natural, Lenguajes de Programación y herramientas de I.A., Creación de un Sistema Experto, Futuro de los Sistemas Expertos... todo un mundo apasionante.

Al final del libro encontraremos una amplia bibliografía que permitirá a los lectores aficionados a este tema profundizar en las entrañas de este futuro cada día más cercano. En cualquier caso, Ingelek nos ofrece una excelente ocasión para ampliar nuestros conocimientos sobre el mundo de la informática. Hay que señalar que no se trata de un «libro de listados» que ofrezca programas para teclear; sin embargo, pensamos que es un libro de interés general que todo amante de la informática debería leer.

Interface

La necesidad de conectar un ordenador con el exterior se pone de manifiesto cuando se desea, por ejemplo, volcar un fichero de texto a una impresora, por no hablar de la transmisión de ficheros de un ordenador a otros a través de modems para línea telefónica o directamente si la distancia lo permite.

En ambos casos se trata de transmisión de datos, aunque la conexión con la impresora no sea un ejemplo específico. Generalmente, la idea de transmisión de datos se asocia en el ámbito del ordenador doméstico, conectado al acoplador acústico y éste al teléfono. No tiene porque ser así en otro entornos, en los que se efectúa la conexión entre ordenadores para formar una red local mediante cable coaxial o fibra óptica, o en la conexión de terminales inteligentes a otros terminales u ordenadores mediante modem a través de línea telefónica dedicada, etc. Hoy en día puede afirmarse que las comunicaciones entre ordenadores son, junto a las tecnologías de fabricación de circuitos integrados, los dos pilares más importantes de la industria del ordenador. Cuando los ordenadores personales comenzaron a tener una gran difusión y la necesidad de conectar estos equipos al una serie de periféricos se hizo patente, muchos fabricantes optaron por utilizar este conocido y probado interface para sus necesidades.

Actualmente, el RS-232C tiene multitud de aplicaciones en el campo de las comunicaciones informatizadas. Entre estas aplicaciones, las más destacadas son la conexión de impresoras, plotters y ordenadores a través de modems, esto sin tener en cuenta la posibilidad de poder usar bases de datos comunitarias, etc.

RS-232 y MSX

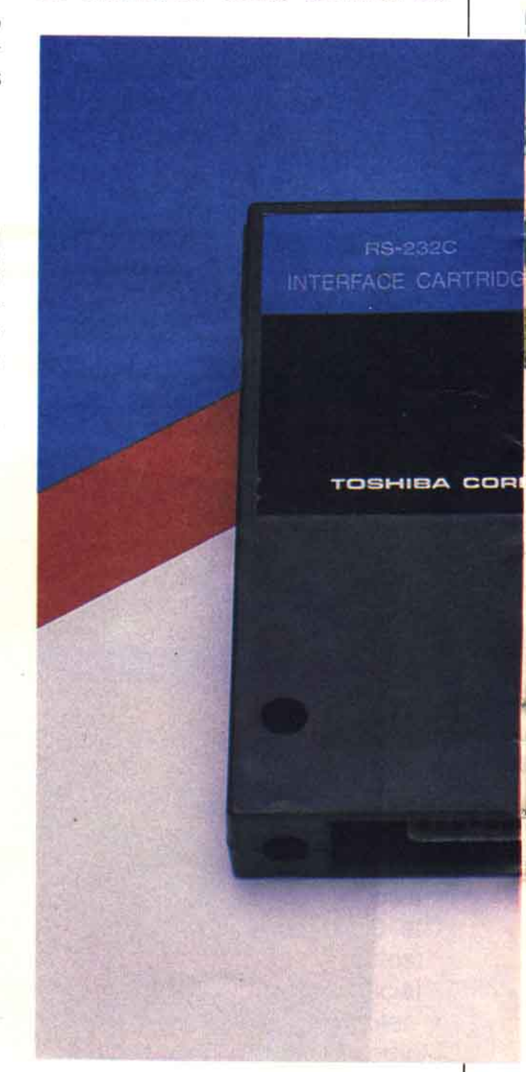
Actualmente, sólo existen en el mercado dos ordenadores MSX que incorporan el interface *RS-232C:* SPECTAVIDEO X'PRESS y el TOSHIBA HX-22. Para el resto de los ordenadores del estándar, se encuentra en el mercado este popular interface, conectable al bus de expansión. Toshiba lo comercializa bajo la denominación HX-R700PE. Además de conectarse al *port* de cartuchos, incorpora una memoria *ROM* que contiene el *software* necesario para su manejo, así como una serie de instrucciones particulares.

A nivel de *BASIC,* se añaden 9 comandos y se amplían otros ya existentes del *MSX BASIC.*

Los nuevos comandos para el control de la comunicación son los siguientes:

CALL COMINI: Utilizando esta instrucción podemos definir todos los parámetros de la comunicación, tales como: velocidad de transmisión y recepción de datos, longitud de palabra, protocolo utilizado y otros parámetros necesarios.

CALL COMTERM: Este comando selecciona el modo terminal, en el cual podremos utilizar nuestro ordenador como terminal de



RS-232C

otro más grande.

CALL COMSTAT: Se utiliza para conocer el estado actual del *interface*. Permite detectar si se está produciendo algún error en la línea de transmisión y el tipo de error de que se trata.

CALL COM GOSUB: Permite especificar el primer número de una rutina de tratamiento de interrupciones generadas por el *interface*. Una interrupción se produce cada vez que el *RS-232C* detecta un intento de transmisión en la línea.

CALL COMON: Habilita las interrupciones.

CALL COMOFF: Deshabilita las interrupciones. Luego existen una serie de instrucciones del *BASIC MSX* que se han ampliado para su utilización con el *RS-232C*. Estas

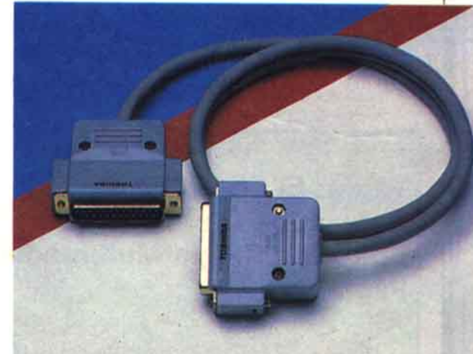


Foto 1: Cable para conectar dos equipos mediante RS-232.

son las más caracteristicas;

SAVE: Envía un programa *BASIC* a través del *RS-232C*. Se utiliza para transmitir programas a otros ordenadores o para listarlos por impresora.

LOAD: Lee un programa *BASIC* desde el *RS-232*. Aunque para recibir correctamente un programa a través del *interface*, debe de estar en formato *ASCII*.

MERGE: Combina un programa recibido desde el *RS-232* con uno existente en la memoria.

OPEN : Abre un canal de entrada o salida de datos al *RS-232C*.

CLOSE : Cierra un canal del *RS-232C*.

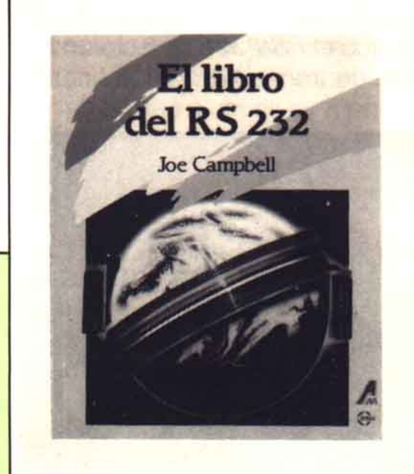
INPUT : Lee un dato desde un canal de entrada.

PRINT : Envía un dato a un canal de salida.

Utilización del RS-232C

Como hemos dicho anteriormente, una de las principales aplicaciones del *RS-232C* es la de conectar un ordenador a un amplio número de dispositivos externos, que generalmente son impresoras o modems (u otros ordenadores). Para el usuario de un MSX, el *RS-232C* supone la posibilidad de conectar su ordenador con gran cantidad de periféricos que no hayan sido diseñados específicamente para el estándar MSX. Otra aplicación interesante consiste en la conexión con el ordenador de algún vecino ya que dadas las características eléctricas del *RS-232C* se pueden conectar dos aparatos a una distancia de unos 160 metros utilizando una velocidad de 4800 baudios con un porcentaje de error menor al de un cassette normal.

Foto 2: Cartucho ROM con software incorporado.



Título: El libro del RS-232
Autor: Joe Campbell
Editorial: Anaya Multimedia
Páginas: 200

El interface *RS-323C,* es el principal medio de conexión entre ordenadores y periféricos. Es poca la literatura escrita sobre este periférico, sin embargo esta obra viene a ocupar y a cumplir la misión de explicar, mostrar y facilitar el entendimiento de este popular interface. El libro constituye una iniciación al *RS-232C,* explicando los principios básicos a un nivel sencillo donde no se necesitan conocimientos técnicos por parte del lector. Comienza con una introducción al *RS-232C,* sus fundamentos y sus aplicaciones. En la primera parte se explican los medios de transmisión de datos (serie, paralelo) y aplicación de la conexión *RS-232C* a los ordenadores. En el segundo capítulo se tratan las bases de la conexión, así como la función de las señales básicas de ésta. El capítulo tercero está dedicado a la *UART* (Universal Asynchronous Receiver Transmitters: Receptores/Transmisores Asíncronos Universales) circuito encargado del trabajo del *RS-232C.* En el capítulo siguiente, encontramos una serie de reglas y trucos para enfrentarnos eficazmente a las conexiones prácticas. A continuación se explican algunos conceptos técnicos necesarios par comprender el funcionamiento y limitaciones del interface *RS-232C.* Estos conceptos están simplificados al máximo, por lo que son fácilmente comprensibles por cualquier persona.

El resto del libro está dedicado al *RS-232C* práctico. En esta parte se describen una serie de herramientas y técnicas para comenzar a realizar conexiones prácticas. También se exponen una serie de casos concretos, detallando todos los pasos a seguir para resolver cada uno de los problemas.

El libro concluye con una especie de catálogo donde se describen algunas de las herramientas y utensilios más utilizados en las prácticas con *RS-232.*

Se trata de un libro adecuado para todos los que deseen adquirir conocimientos sobre el *RS-232C,* así como para los que simplemente deseen conectar una impresora o cualquier otro tipo de periférico. El lenguaje utilizado es muy aseguible y contiene numerosos y sencillos ejemplos por lo que su lectura resulta entretenida y sencilla.


GANE
7.000 PTAS.
todos los meses
PARTICIPANDO EN NUESTRO CONCURSO
MSX Magazine premiará cada mes los programas que nos hagan llegar nuestros lectores.
Para participar en este concurso abierto, todo aficionado a los ordenadores con este estándar deberá hacer llegar a la redacción de la revista el listado, un cassette y un texto explicativo.
Entre todos los programas que recibamos cada mes, serán seleccionados para su publicación aquellos que reúnan los siguientes criterios:
— Originalidad de la aplicación.
— Simplicidad del método de programación.
La única condición para participar en el concurso será que los programas no hayan sido publicados previamente en ninguna revista.
Enviar vuestros programas a: MSX Magazine
C/Bravo Murillo, 377 - 5.° A 28020 MADRID
BANCO DE ESPAÑA
MIL pesetas
CINCO MIL pesetas
2E1554046
14457150

Utilización de ficheros de datos



En el último número de **MSX Magazine** vimos un ejemplo de utilización de variables con subíndices. Con el programa que veíamos entonces podíamos almacenar en la memoria del ordenador los nombres, direcciones y teléfonos de nuestros amigos. Pero de poco nos servía todo eso ya que, una vez desconectado el ordenador, toda la información almacenada se perdía. BASIC MSX posee, sin embargo, un conjunto de instrucciones que nos permitirá guardar en cinta magnetofónica toda esa información y reutilización cada vez que nos interesa.

Seguiremos con el programa del listín telefónico, que iremos engrosando y mejorando con las nuevas instrucciones y características de Basic MSX que vayamos aprendiendo.

Cómo abrir un fichero

En todo lo que sigue nos limitaremos al uso de ficheros en cinta magnetofónica. Hablaremos de los ficheros en disco cuando la serie está más avanzada, ya que, casi con toda seguridad, los usuarios que poseáis una unidad de disco seréis relativamente pocos. No obstante, todo lo dicho aquí es válido para ficheros en disco de acceso secuencial.

El sistema MSX nos permite pasar datos de la memoria principal a cinta magnetofónica (y, lógicamente, retornar los datos de la cinta magnética almacenándola en la memoria principal). Pero para ello debemos ABRIR un archivo, de la misma forma que para escribir o leer en un listín telefónico los teléfonos y direcciones de nuestros amigos tenemos que abrir el listín.

MSX nos permite abrir hasta 15 ficheros simultáneamente, pero para ello tenemos que indicarle al ordenador cuántos archivos vamos a utilizar. Esto se hace con la sentencia *MAXFILES,* como sigue:

```
10 MAXFILES N
```

en que N puede valer de 0 a 15. Es importante, cuando utilicemos esta sentencia, hacerlo al principio del programa (lo mismo que la instrucción *CLEAR*) ya que este tipo de sentencias provocan una reordenación interna de la memoria.

No obstante, lo dicho anteriormente, si n es igual a 1, puede omitirse la sentencia *MAXFILES,* ya que 1 es el valor que se toma por defecto al conectar el ordenador. (Cuando n=0 no podemos utilizar ningún fichero, pero se gana algo de memoria).

Podemos abrir un archivo para dos cosas: para escribir datos en cinta magnética, o para leer datos anteriomente escritos en cinta. Para pasar datos del ordenador a la cinta utilizaremos la sentencia.

```
1000 OPEN "NOMBRE" FOR OUTPUT AS N
```

En esta sentencia, "NOMBRE" es el nombre que queremos dar al archivo. Debe ir entre comillas y tendrá como máximo 6 dígitos, de los que el primero debe ser una letra, y N es el número del archivo que queremos abrir, que estará dentro del rango marcado por la sentencia MAXFILES al principio del programa. (Si no hemos ampliado el número de archivos disponibles con la sentencia *MAXFILES,* entonces, necesariamente, N=1).

Si lo que queremos es leer da-



tos de una cinta magnética, escribiremos la sentencia.

1000 OPEN "NOMBRE" FOR INPUT AS N

que se diferencia de la anterior en que aquí hemos escrito *INPUT* (entrada) en lugar de *OUTPUT* (salida).

Cómo cerrar un archivo

Hemos visto que (siempre en relación con la utilización de cinta magnética como soporte para almacenar datos) podemos abrir un archivo para escribir en él o para leer de él, pero no podemos hacer las dos cosas a la vez. Por eso, después de haber escrito en el archivo 1 (teniendo, por defecto, *MAXFILES*=1), no podremos leer en él si no lo cerramos préviamente. En cualquier caso es conveniente cerrar todos los archivos abiertos en un programa antes de abandonar éste, ya que en caso contrario (salvo si desconectamos el ordenador o cerramos los archivos en modo directo) no podremos abrir ese archivo nuevamente. (Obtendremos un error del tipo *FILE ALREADY OPEN,* Archivo ya abierto).

Para cerrar un archivo abierto con la sentencia OPEN, utilizaremos la sentencia *CLOSE* de una de las tres formas siguientes:

CLOSE N – Cierra el archivo abierto con el número N.

CLOSE N, O, P... – Cierra los archivos abiertos con los números N, O, P...

CLOSE – Cierra todos los archivos abiertos anteriormente.

Escribir y leer un archivo

No vamos a ver, por ahora, todas las posibles formas de escribir o leer un archivo en cinta magnética, sino solamente la forma más útil para escribir y las dos formas más útiles para leer.

La forma más sencilla de escribir datos numéricos (sean del tipo que sean) y alfanuméricos es mediante la instrucción *PRINT N,* que significa ESCRIBIR en el archivo *N,* por ejemplo:

```
200 PRINT  1, A%
300 PRINT  1, A!
400 PRINT  1, A
500 PRINT  1, A$
```

NOTA: Aunque es posible escribir en un archivo varias variables con una sola instrucción *PRINT N,* (*PRINT 1,* A%, A!, A , A$) recomendamos no hacerlo así, pues podrían originarse errores de lectura.

Para leer variables numéricas de un archivo, lo haremos mediante la instrucción *INPUT N,* seguida del nombre de la variable:

```
600 INPUT  1, A%
700 INPUT  1, A!
800 INPUT  1, A
```

Cuando se trata de leer variables alfanuméricas, la forma más sencilla y útil de hacerlo es con la instrucción *LINE INPUT N,* seguida del nombre de la variable.

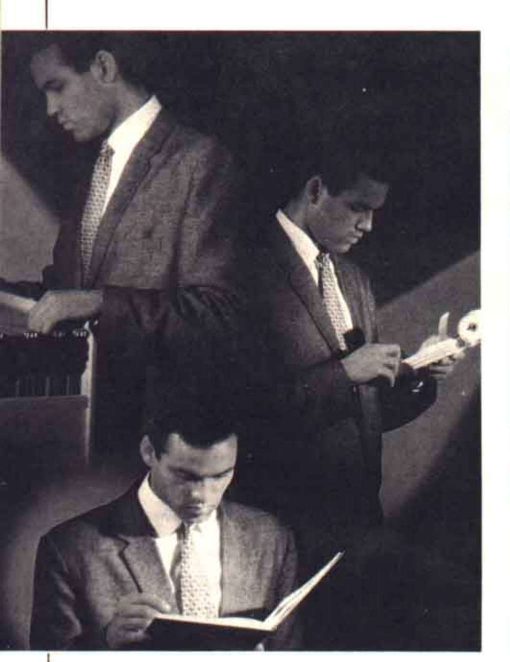
```
900 LINE INPUT  1, A$
```

(esta instrucción es muy útil, ya que lee variables alfanuméricas hasta que encuentra un código de retorno, es decir, un indicativo de que la variable ha terminado. Este código se introduce automáticamente al grabar el dato en la cinta mediante la instrucción *PRINT N,*).

Veamos ahora cómo queda modificado el programa 2 del mes pasado al introducir un fichero para escribir y otro para leer los datos personales de nuestros amigos. (Véase el diagrama de flujo y el programa listado en estas páginas).

Al principio el programa es igual que el programa 2 del mes pasado: hemos determinado el espacio de memoria asignado a variables alfanuméricas y el modo de pantalla, dimensionando la variable *A$* y definido la variable *B$.* Entre las líneas 110 y 270 hemos introducido un pequeño «menú» que nos permitirá elegir la opción que deseamos ejecutar (introducir fichas en memoria, leerlas, o recuperar el archivo previamente grabado en cinta).

Si elegimos la opción 1, se ejecutará el trozo de programa comprendido entre las líneas 280 y



I
DIM A$ (100.2)
MENU:
1. INTRODUCIR FICHAS
2. LEER FICHAS
3. RECUPERAR DATOS DE CINTA.
A%
A%
=1
=2
=3
INTRODUCIR FICHAS
¿OTRO NUMERO? A$
A$="S"
SI
NO
¿GRABAR CINTA? A$
A$="S"
NO
SI
GRABAR
¿VUELTA AL MENU? A$
A$="S"
SI
NO
F
¿NOMBRE? A$
LEER FICHA A$
¿OTRA FICHA? A$
A$="S"
SI
NO
TOMAR DATOS DE CINTA

630. Las líneas 320 y 440 son similares a las correspondientes del programa 2 y nos permiten introducir los datos en la memoria del ordenador. Al terminar una ficha, podemos elegir entre introducir una nueva ficha o no hacerlo. En este caso (línea 450), el programa nos preguntará si queremos grabar el fichero en cinta magnética. Si no es así, el programa vuelve al menú, y en caso contrario se ejecutan las líneas 460 a 630. En la línea 520 se abre el archivo con el nombre «TELEFONOS» (las tres últimas letras «NOS» son ignoradas por el ordenador) para escribir en cinta *(OUTPUT),* y a continuación se escriben los datos mediante la sentencia *PRINT 1,* según hemos indicado. Una vez terminado de grabar, el archivo se cierra (línea 590), y el programa nos pregunta si queremos volver al menú (línea 610), si la respuesta es distinta de «S» (línea 620) el programa termina (sentencia *END* de la línea 630). (Si por error hemos salido del programa, podemos continuar escribiendo —en modo directo— *GOTO* 150).

La opción 2 nos permite leer las fichas de la misma forma que se hacía en el programa 2, y con la opción 3 podemos recuperar los datos de cinta, para escribir más datos o simplemente para leerlos. El funcionamiento de este trozo de programa (líneas 640 a 800) es parecido al de grabación. La diferencia está en que aquí vamos a leer en lugar de escribir, y, por tanto, abrimos el archivo (línea 710) para leer *(INPUT).* La sentencia *OPEN... FOR OUTPUT...* escribe una cabecera de archivo que contiene el nombre, de forma que dicho archivo pueda ser identificado por la sentencia *OPEN... FOR INPUT...* Para saber que dicha cabecera ha sido encontrada hemos puesto la instrucción *PRINT* de la línea 720. A continuación se lee la variable numérica *I%* (contador) y acto seguido se leen las fichas (variables alfanuméricas *A$*(I1%,J%) de la línea 760). Una vez cerrado el archivo, el programa vuelve al menú.

J. Antonio Castejón

```
10 REM LISTIN TELEFONICO - 3
20 REM ====================
30 REM
40 CLEAR 20000
50 SCREEN 0,,0
60 DIM A$(100,2)
70 B$(0)="NOMBRE   "
80 B$(1)="DIRECCION"
90 B$(2)="TELEFONO "
100 REM
110 REM
120 REM MENU
130 REM ----
140 REM
150 CLS
160 PRINT "                    MENU"
170 PRINT "                    ----"
180 PRINT
190 PRINT "PULSA 1 PARA INTRODUCIR FICHA
     S"
200 PRINT "PULSA 2 PARA LEER FICHAS"
210 PRINT "PULSA 3 PARA RECUPERAR DATOS
     DE CINTA"
220 PRINT
230 INPUT A%
240 IF A%=1 THEN 320
250 IF A%=2 THEN 850
260 IF A%=3 THEN 680
270 GOTO 150
280 REM
290 REM INTRODUCIR FICHAS
300 REM -----------------
310 REM
320 CLS
330 I%=I%+1
340 PRINT "FICHA Nº";I%
350 FOR I1%=0 TO 2
360 PRINT
370 PRINT B$(I1%);
380 INPUT A$(I%,I1%)
390 NEXT I1%
400 PRINT
410 PRINT
420 INPUT "¿DESEAS INTRODUCIR OTRO NUMER
     O...S/N";A$
430 IF A$="S" THEN 320
440 PRINT
450 INPUT "¿DESEAS GRABAR EN CINTA......
     ....S/N";A$
460 IF A$="N" THEN 150
470 PRINT
480 INPUT "SI LA CINTA ESTA PREPARADA PU
     LSA 'S'";A$
490 IF A$<>"S" THEN 480
500 PRINT
510 PRINT "GRABANDO..."
520 OPEN "TELEFONOS" FOR OUTPUT AS#1
530 PRINT#1,I%
540 FOR I1%=1 TO I%
550 FOR J%=0 TO 2
560 PRINT#1,A$(I1%,J%)
570 NEXT J%
580 NEXT I1%
590 CLOSE #1
600 PRINT
610 INPUT "¿DESEAS VOLVER AL MENU.......
     ....S/N";A$
620 IF A$="S" THEN 150
630 END
640 REM
650 REM RECUPERAR DATOS DE CINTA
660 REM ------------------------
670 REM
680 CLS
690 INPUT "SI LA CINTA ESTA PREPARADA PU
     LSA 'S'";A$
700 IF A$<>"S" THEN 680
710 OPEN "TELEFONOS" FOR INPUT AS#1
720 PRINT "ARCHIVO ENCONTRADO..."
730 INPUT#1,I%
740 FOR I1%=1 TO I%
750 FOR J%=0 TO 2
760 LINE INPUT#1,A$(I1%,J%)
770 NEXT J%
780 NEXT I1%
790 CLOSE #1
800 GOTO 150
810 REM
820 REM BUSCAR FICHA
830 REM ------------
840 REM
850 CLS
860 INPUT "DIME EL NOMBRE";A$
870 FOR I1%=1 TO I%
880 IF A$<>A$(I1%,0) THEN NEXT I1%
890 CLS
900 FOR I2%=0 TO 2
910 PRINT
920 PRINT B$(I2%);": ";A$(I1%,I2%)
930 NEXT I2%
940 PRINT
950 PRINT
960 PRINT
970 INPUT "¿DESEAS BUSCAR OTRA FICHA....
     .S/N";A$
980 IF A$="S" THEN 850
990 GOTO 150
```

BASIC en Código Máquina

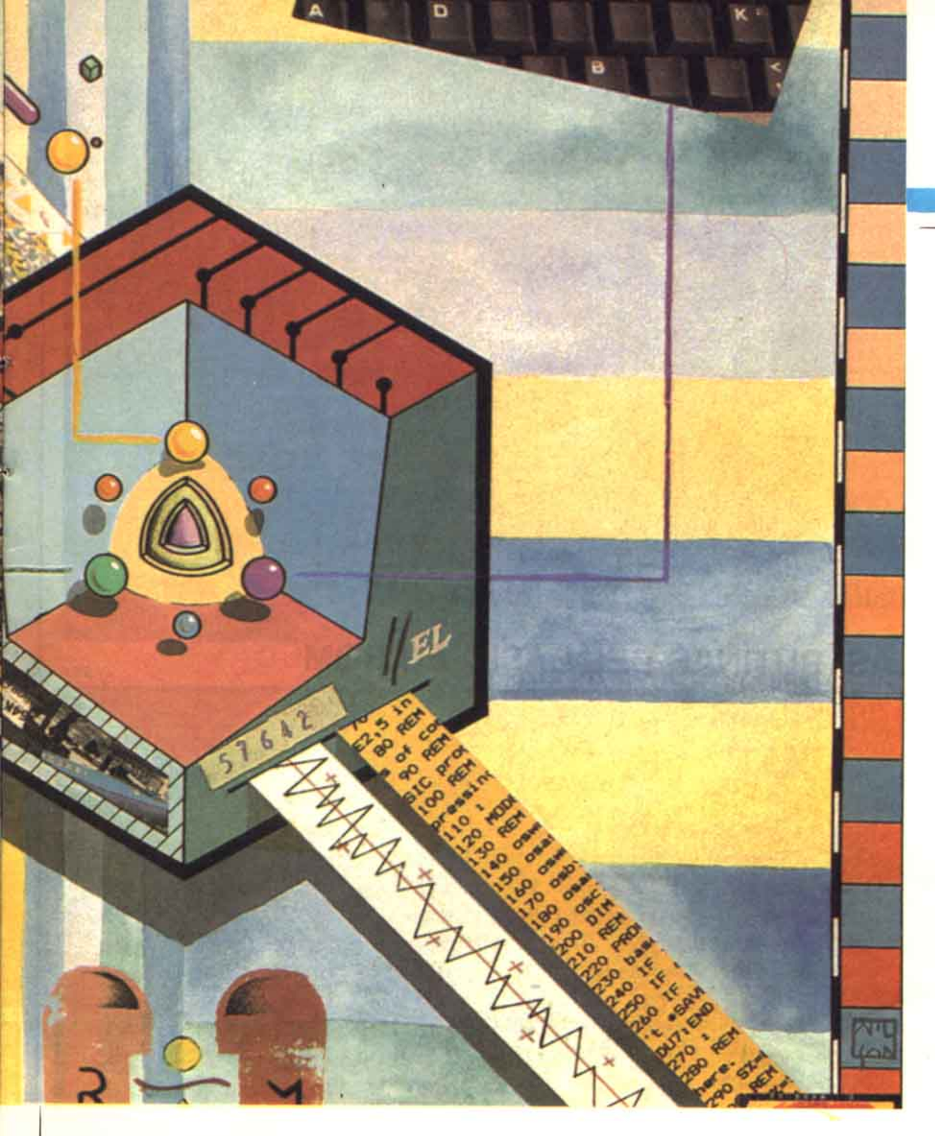
No es objeto de este artículo el de convencer a los lectores de las enormes ventajas y posibilidades que ofrece la programación en Código Máquina (C/M), ni tampoco la de advertir de los grandes inconvenientes que al programador medio se le presentan a causa de la dificultad que plantea su correcta utilización. El *BASIC* MSX es cómodo de manejar, sencillo de utilizar y proporciona una gran diversidad de comandos y funciones que facilitan enormemente la realización de cualquier programa. Su diversidad de tipos de variables, de funciones matemáticas definidas, de funciones de presentación en pantalla y su facilidad para el tratamiento de ficheros, son elementos que, añadidos a los macrolenguajes de música y gráficos, instrumentalizan más que adecuadamente el ingenio y capacidad de programación de cualquier usuario de este lenguaje. Pero no todo van a ser ventajas, y al programador en *BASIC* MSX se le presentan básicamente dos graves inconvenientes: la lentitud en la ejecución y la gran ocupación de memoria que impone este lenguaje.

Frente a estos problemas, y como la mejor solución a ellos, se abre la posibilidad de programar en *C/M,* lenguaje que los soluciona gracias a su gran rapidez y mínima ocupación de memoria, además de su potencia, que le permite acceder a todos los recursos que le brinda el estándar MSX. Pero como ya indicábamos al comienzo de este artículo, tampoco son pequeños los problemas que el *C/M* plantea al programador, aunque de otra índole a los del *BASIC.* Fundamentalmente, podemos reducirlos uno, y es su gran complejidad, que a su vez genera dificultad en el entendimiento y lentitud en la programación. A ello hay que añadir otro de gran importancia: no tiene todos los comandos y funciones que tiene el *BASIC* MSX.

En este artículo sólo se prentende indicar lo que podría ser una solución al último de los problemas planteados, es decir, la posibilidad de que programas que requieran gran velocidad en su ejecución por su propia naturaleza (de juegos o con muchos cálculos y bucles) puedan ser realizados en *C/M* utilizando todos los comandos y funciones del *BASIC,* sin necesidad de crear, con gran dificultad, las rutinas que ya se encuentran en la *ROM.*

A medida que un programador va introduciendo en el micro un programa en *BASIC,* aquél, aunque lo almacena todo en la memoria, con sus comandos y sus funciones, no lo hace de la misma manera para todos sus elementos. Para ahorrar memoria y aumentar la velocidad de ejecución, el ordenador selecciona las palabras reservadas para los comandos y funciones (palabra-clave) y las guarda sólo como uno o dos *bytes.* Así, por ejemplo, una línea del programa que contuviese la siguiente orden: PRINT VAL("MSX") sería guardada en memoria como sigue:

91,20,FF,94,28,22,4D,53,58,22 y 29 (todo ello en hexadecimal)d. Para *PRINT* (comando) la memoria es ocupada con un solo *byte,* el correspondiente a su código equivalente: 91. El 20 sería el espacio entre *PRINT* y *VAL,* y el resto sería el equivalente a los parámetros del comando: *VAL("MSX")*>. De todos ellos, la función *(VAL)* es representada simplemente por su



código (FF,94), mientras que todo lo que resta son los códigos *ASCII* de «MSX». Es decir, una línea compuesta por los cinco *bytes* de los códigos *ASCII* del comando *(PRINT)* y diez de los parámetros *<VAL("MSX")>*, se registra en la memoria del ordenador como nueve *bytes*, de los cuales tres corresponden a códigos predefinidos por el *BASIC* para las palabras-clave, ya sean comandos o funciones (*PRINT* y *VAL*, respectivamente).

El proceso que se verifica, en la ejecución de dicha línea por el intérprete de *BASIC* es, esquemáticamente, el siguiente: el *BASIC* apunta con el registro HL al primer *byte* de la cadena que compone los parámetros del comando *PRINT*, que será el correspondiente a la función *VAL*, es decir, a la dirección donde se encuentra el FF. A continuación llama (realiza un *CALL*) a la rutina de la *ROM* que ejecuta al comando *PRINT*. Cuya ejecución incide sobre los parámetros apuntados previamente por HL. Para el comando *PRINT* la rutina se encuentra en &H4A24.

Cuando se ejecute una de estas rutinas desde *C/M* hay que tener cuidado de acabar siempre la cadena de parámetros con dos puntos «:» para evitar la realización de algún error. Por ejemplo: *<VAL("MSX"):>*.

La tabla adjunta contiene todos los comandos y funciones del *BASIC* con sus códigos y la dirección de la *ROM* donde están las rutinas que ejecuta cada uno de ellos.

De todas las facilidades que esas rutinas del *BASIC* nos dan, hemos elegido los comandos *PLAY* y *DRAW* para desarrollar unas rutinas por considerarlos interesantes para los lectores.

El comando *PLAY* utiliza una cadena alfanumérica que contiene el macrolenguaje de música. Para ejecutarla desde *C/M* hay que entrecomillarla y a continuación de ella, como ya se dijo, se han de poner dos puntos.

La potente rutina de música del *BASIC* es fácilmente extrapolable al ensamblador:

```
         LD    HL,CADENA ; Apunta a
                           los parámetros.
         CALL  $73E5       ; Ejecuta el
                             PLAY
         RET
CADENA  :DB     '"V10O4CDEF":'
```

Para utilizar los tres canales de música, se realiza lo mismo y las cadenas se separan por comas (igual que en *BASIC*).

```
         LD    HL,CADENA
         CALL  $73E5
         RET
CADENA  :DB    '"01CDEFGG",'
         DB    '"04DEFGFA",'
         DB    '"V6EFGABA":'
```

Para los gráficos, por su parte, primero hay que ejecutar *SCREEN 2* para luego utilizar el *DRAW*.

```
         LD    HL,SCREEN  ; Apunta el
                 parámetro de SCREEN.
         CALL  $79CC       ; Rutina en
ROM de SCREEN.
         LD    HL,CADENA  ; Apunta a
                 la cadena de gráficos.
         CALL  $5D6E       ; Llama a
DRAW.
         RET
SCREEN  :DB    $13,':'
CADENA  :DB    '"BM100,150U50E50'
               'F50D50L100":'
```

Otra particularidad del *BASIC* es que las constantes numéricas tampoco se guardan como carácteres ASCII, sino que lo hace de una forma más complicada según el tipo de variable que sea. Dejamos a la inteligencia del lector el que descubra de qué forma se guardan estas constantes.

Si no se está seguro de cómo guarda el *BASIC* los parámetros de algunos comandos. Lo mejor que se puede hacer es escribir la línea en *BASIC* para luego mirar la memoria con *pokes*. Si lo escribe en la primera línea del programa y tiene un ordenador de 32 ó 64 K *RAM*, el comando se encuentra en la dirección &H8005.

J. Domingo Sandoval



TABLA DE LAS RUTINAS DEL BASIC EN ROM

Palabra	código	dirección	palabra	código	dirección	Palabra	código	dirección	palabra	código	dirección
END	81	63EA	FOR	82	4524	TIME	CB	7911	KEY	CC	786C
NEXT	83	6527	DATA	84	485B	MAX	CD	7E4B	MOTOR	CE	73B7
INPUT	85	4B6C	DIM	86	5E9F	BLOAD	CF	6EC6	BSAVE	D0	6E92
READ	87	4B9F	LET	88	4880	DSKO$	D1	7C16	SET	D2	7C1B
GOTO	89	47E8	RUN	8A	479E	NAME	D3	7C20	KILL	D4	7C25
IF	8B	49E5	RESTORE	8C	63C9	IPL	D5	7C2A	COPY	D6	7C2F
GOSUB	8D	47B2	RETURN	8E	4821	CMD	D7	7C34	LOCATE	D8	7766
REM	8F	485D	STOP	90	63E3	USR	DD	4FD5	FN	DE	5040
PRINT	91	4A24	CLEAR	92	64AF	ERL	E2	4E0B	STRING$	E3	6829
LIST	93	522E	NEW	94	6286	INSTR	E5	68EB	VARPTR	E7	4E41
ON	95	48E4	WAIT	96	401C	CSRLIN	E8	790A	ATTR$	E9	7C43
DEF	97	501D	POKE	98	5423	DSKI$	EA	7C3E	INKEY$	EC	7347
CONT	99	6424	CSAVE	9A	6FB7	POINT	ED	5803	LEFT$	FF 81	6861
CLOAD	9B	703F	OUT	9C	4016	RIGHT$	FF 82	6891	MID$	FF 83	689A
LPRINT	9D	4A1D	LLIST	9E	5229	SGN	FF 84	2E97	INT	FF 85	30CF
CLS	9F	00C3	WIDTH	A0	51C9	ABS	FF 86	2E82	SQR	FF 87	2AFF
ELSE	A1	485D	TRON	A2	6438	RND	FF 88	2BDF	SIN	FF 89	29AC
TROFF	A3	6439	SWAP	A4	643E	LOG	FF 8A	2A72	EXP	FF 8B	2B4A
ERASE	A5	6477	ERROR	A6	49AA	COS	FF 8C	2993	TAN	FF 8D	29FB
RESUME	A7	495D	DELETE	A8	53E2	ATN	FF 8E	2A14	FRE	FF 8F	69F2
AUTO	A9	49B5	RENUM	AA	5468	INP	FF 90	4001	POS	FF 91	4FCC
DEFSR	AB	4718	DEFINT	AC	471B	LEN	FF 92	67FF	STR$	FF 93	6604
DEFSNG	AD	471E	DEFDBL	AE	4721	VAL	FF 94	68BB	ASC	FF 95	680B
LINE	AF	4B0E	OPEN	B0	6AB7	CHR$	FF 96	681B	PEEK	FF 97	541C
FIELD	B1	7C52	GET	B2	775B	VPEEK	FF 98	7BF5	SPACE$	FF 99	6848
PUT	B3	7758	CLOSE	B4	6C14	OCT$	FF 9A	65F5	HEX$	FF 9B	65FA
LOAD	B5	6B5D	MERGE	BC	6B5E	LPOS	FF 9C	4FC7	BIN$	FF 9D	65FF
FILES	B7	6C2F	LSET	B8	7C48	CINT	FF 9E	2F8A	CSNG	FF 9F	2FB2
RSET	B9	7C4D	SAVE	BA	6BA3	CDBL	FF A0	303A	FIX	FF A1	30BE
LFILES	BB	6C2A	CIRCLE	BC	5B11	STICK	FF A2	7940	STRIG	FF A3	794C
COLOR	BD	7980	DRAW	BE	5D6E	PDL	FF A4	795A	PAD	FF A5	7969
PAINT	BF	59C5	BEEP	C0	00C0	DSKF	FF A6	7C39	FPOS	FF A7	6D39
PLAY	C1	73E5	PSET	C2	57EA	CVI	FF A8	7C66	CVS	FF A9	7C6B
PRESET	C3	57E5	SOUND	C4	73CA	CVD	FF AA	7C70	EOF	FF AB	6D25
SCREEN	C5	79CC	VPOKE	C6	7BE2	LOC	FF AC	6D03	LOF	FF AD	6D14
SPRITE	C7	7A48	VDP	C8	7B37	MKI$	FF AE	7C57	MKS$	FF AF	7C5C
BASE	C9	7B5A	CALL	CA	55A8	MKD$	FF B0	7C61			

Todos los códigos de las palabras reservadas del *BASIC* y las direcciones de las rutinas en *ROM* están en hexadecimal

El BIOS de la memoria de vídeo

(y VIII)

Para los no inciados en Código Máquina, este artículo puede ser un barullo incomprensible, pero si prestáis un poco de atención no es demasiado complicado. Los únicos conocimientos que necesitáis para entenderlo son:

— Conocer la existencia de los registros A, F, HL, BC, DE.

— Saber que la *RAM* tiene 65536 octetos, de los cuales vamos a tener que utilizar unos cuantos.

— Saber que el Código Máquina se basa en una larga serie de instrucciones llamadas «nemónicos», una de las cuales *(«CALL»)* sirve para acudir al *BIOS*.

¿Qué es el BIOS?

En pocas palabras, el *BIOS (Basic Input-Output System* = Sistema Básico de Entrada y Salida) es una larga lista de subrutinas en Código Máquina que el programador puede emplear siempre que quiera, y gracias a las cuales puede lograr ciertos objetivos, en la mayor parte de los casos inalcanzables dese *BASIC*, muy rápidamente. Estas subrutinas están numeradas en hexadecimal, de tres en tres. Para que se ejecute una subrutina, sólo hay que emplear la orden *CALL* en Código Máquina, y a continuación el número de la subrutina en hexadecimal.





Con este artículo cerramos la serie que hemos dedicado a la memoria de vídeo. Esperamos que pronto invadan nuestra redacción miles de programas de buena calidad, empleando los muchísimos mecanismos gráficos que hemos ido desarrollando.

el bios

De esa larga lista nosotros sólo vamos a hablar de aquellas que se refieren, de una manera u otra, a la memoria de vídeo. Muchas de ellas no son de aplicación inmediata, es decir, se necesitan unas a otras para lograr algún objetivo. Por culpa de esto, no podremos comentar muchas aplicaciones, por lo que quizá este artículo os parezca mucho menos útil que los otros siete, poro lo cierto es que sí son útiles: ya iréis viendo su utilidad con la práctica.

Las subrutinas del BIOS

La primera de las subrutinas del *BIOS* de la memoria de vídeo tiene el número hexadecimal &H41. Por lo tanto, para que funcione tenéis que programar *CALL* &H0041 (fijaos que se ponen cuatro cifras hexadecimales, porque si no, el ordenador no lo entiende). Lo que hace esta subrutina es lo mismo que sucede cuando ponéis a 0 el *bit BLANK*, del registro *VDP*(1), como veíamos en el número anterior. La pantalla pierde todo el color de la tinta, y se pone del color de fondo. Sus aplicaciones son las mismas que en aquel caso. Esta subrutina funciona combinada con la siguiente, y recibe el nombre de *DISSCR*.

La siguiente, la &H0044, recibe el nombre de *ENASCR*, y su función es la inversa, es decir, volver a pintar en la pantalla los colores que ha borrado la ejecución de la &H0041, y poner también en la pantalla todo aquello que haya sido introducido después desde teclado. Se suele utilizar junto con la anterior para crear parpadeos o apariciones instantáneas en la pantalla. Es equivalente a poner a 1 el *bit BLANK*.

La subrutina que va a continuación es la &H0047, que recibe el nombre de *WRTVDP*, y sustituye a la orden en *BASIC: VDP*(a)=b, donde b es un número entre 0 y 255, y a es el registro *VDP* donde se va a introducir ese número. El valor de a se introduce en el registro C, y el valor de b en el registro B (B y C son la pareja de registros del microprocesador *280A)*. A continuación se llama a esta subrutina y suceden dos cosas: la primera es que el registro *VDP*(a) se introduce el valor b, y la segunda es que en el octeto (&HF3DF + a) de la memoria *RAM* se introduce el número b, porque si luego necesitáis saber cuál era el número b, no tenéis más que mirar en el octeto &HF3DF + a, y lo sabréis. Por ejemplo, si en *VDP*(3) queréis intodűcir un 1, con esta subrutina lográis que se introduzca ese 1 en *VDP*(3), y que además se introduzca un 1 en el octeto &HF3DF + 3, es decir, en el octeto &HF3E2.



La subrutina número &H004A recibe el nombre de *RDVRM*, y sirve para que el programador pueda conocer el número que hay en un octeto de la memoria de vídeo. Sustituye a la sentencia *VPEEK* en *BASIC*. Si, por ejemplo, queréis saber qué número contiene el octeto 6192 de la memoria de vídeo, tendréis que meter en HL el número 6192. Tras la ejecución de esta subrutina, el número que estáis buscando aparecerá en A.

La subrutina siguiente, la &H004D, recibe el nombre de *WRTVRM*, y sustituye al *VPOKE* a,b del *BASIC*. Su función es, por lo tanto, la inversa de la de la subrutina anterior, es decir, introducir un número b en un octeto a de la memoria de vídeo. Si introducimos a en HL y b en A, y a continuación ejecutamos la subrutina, ya lo habremos logrado.

La siguiente, la subrutina &H0050, recibe el nombre de *SETRD*. No tendréis que usarla con frecuencia, pues su única función es ayudar a leer de la memoria de vídeo. Si queréis leer algo de la memoria de vídeo, hay otras subrutinas en el *BIOS* que lo leen directamente, como la &H0041, que ya hemos visto. La subrutina *SETRD* sólo conviene utilizarla en el caso de que queramos establecer un contacto directo con la memoria de vídeo. En caso de que la queramos utilizar, se introduce el octeto a leer en el registro HL, y una vez ejecutada la subrutina, ya estará dispuesta la memoria de vídeo a que la leamos.

La subrutina &H0053, llamada *SETWRT*, es muy semejante a la anterior, con la única diferencia de que ésta se refiere a introducir un número en un octeto. Tampoco es

SERVICIO DE EJEMPLARES ATRASADOS

ESTOS SON LOS EJEMPLARES DE MSX MAGAZINE APARECIDOS EN EL MERCADO CON UN RESUMEN DE SU CONTENIDO



Núm. 1
¿Qué es el MSX? Su BASIC, periféricos, programas, software.



Núm. 4
Las comunicaciones entre ordenadores, la jerga informática, trucos, rincón del lector.



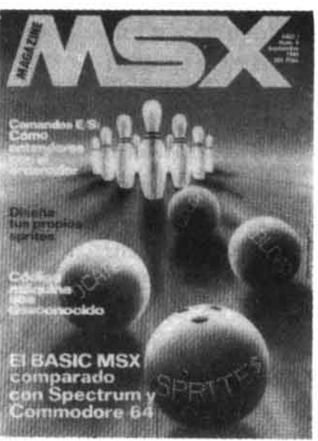
Núm. 6
Los 8 magníficos (test gigante), el bus de expansión, los misterios de la grabación, programas.



Núm. 8
Compact Disc, el periférico del futuro. Test: Dynadata DPC-200. Continuamos con la memoria de video. Libros, software, programas, trucos.



Núm. 2
Generación de sonido. MSX-DOS, el ordenador por dentro, programas, noticias.



Núm. 5
Comandos de entrada/salida, el BASIC MSX comparado con Spectrum y Commodore 64. Código Máquina.



Núm. 7
Analizamos el Generador de Sonido. Aplicaciones matemáticas con el ordenador. La memoria de video. Trucos, noticias.



Núm. 9
Características técnicas del Compact Disc. Tratamiento de datos. Test: Quick Disk. Trucos, libros, noticias, programas.



Núm. 3
Los joysticks, 256 caracteres programables, Z80 corazón de león, compro/vendo/cambio.

PARA HACER SU PEDIDO, RELLENE ESTE CUPON, HOY MISMO Y ENVIELO A MSX MAGAZINE BRAVO MURILLO, 377. Tel. 733 96 62 - 28020 MADRID

Ruego me envíen los siguientes números atrasados ....

al precio de **300** ptas. cada uno. Cuyo importe abonaré:

☐ POR CHEQUE ☐ CONTRA REEMBOLSO ☐ CON MI TARJETA DE CREDITO

☐ AMERICAN EXPRESS ☐ VISA ☐ INTERBANK

Número de mi tarjeta ....

Fecha de caducidad ....

NOMBRE ....

DIRECCION ....

POBLACION .... C.P. ....

PROVINCIA ....

necesario emplearla frecuentemente; el octeto en el que tendríamos que escribir se introduce en el registro HL.

La subrutina &H0056 recibe el nombre de *FILVRM.* Al igual que la subrutina &H004D, sirve para introducir datos en la memoria de vídeo, con la peculiaridad de que esta subrutina introduce el mismo número en muchos octetos a la vez. Por ejemplo, si queréis hacer una figura móvil en forma de cuadrado, de 16 x 16, tendréis que introducir 255 en los octetos del 14336 al 14367, ocupando, pues, 32 octetos. Utilizando esta subrutina tendréis que introducir en A, 255, en BC, 32 (el número de octetos que ocupamos), y en HL, 14336, que es el primer octeto ocupado.

La subrutina &H0059, también llamada *LDIRMV,* desplaza un bloque de datos desde la memoria de vídeo a la *RAM.* Para utilizarla, tenéis que introducir en HL, el primer octeto de la memoria de vídeo donde comienza la información, en DE, el primer octeto de la *RAM* a donde se desplaza ese bloque de datos, y en BC, la cantidad de datos que tiene ese bloque. Su aplicación principal es poder almacenar pantallas en una cinta, mediante la orden en *BASIC"-BLOAD",* que carga a cassette bloques de la *RAM.*

La subrutina &H005C, llamada *LDIRWM,* es la opuesta a la anterior, y desplaza bloques de datos de la *RAM* a la memoria de vídeo. En HL tenéis que introducir el primer octeto de la *RAM* (y no de la memoria de vídeo, como en la subrutina anterior) en el que comienza el bloque, en DE el primer octeto de la memoria de vídeo que recibe el bloque, y en BC la cantidad de datos que hay en el bloque.

La subrutina &H005F es la equivalente a la orden en *BASIC "SCREEN",* con su primer parámetro, es decir, cambia el modo de pantalla donde nos hallemos. El único dato que tenemos que darle es el modo *SCREEN* que deseamos (0, 1, 2, ó 3). Introduciendo uno de esos cuatro números en A y ejecutando la subrutina, habremos cambiado el modo *SCREEN.* Esta subrutina recibe el nombre de *CHGMOD.*

La subrutina número &H0062, llamada *CHGCLR,* sirve para establecer los colores de tinta, de fondo y de borde en el modo de pantalla en el que estemos. Pero los datos que hemos de introducir en esta subrutina no los metemos en los registros, sino en octetos de la *RAM.* En el octeto de la *RAM* número &HF3E9, hemos de introducir el nuevo color de la tinta, en el &HF3EA el color de fondo, y en el &HF3EB el color de borde. Si a continuación ejecutamos la subrutina, aparecerán esos colores en la pantalla.

La subrutina &H0069 recibe el nombre de *CLRSPR,* y sirve para borrar todo lo que la memoria de vídeo tuviera almacenado sobre figuras móviles. Esta subrutina actúa a la vez sobre las tablas 8,9,13,14,18 y 19. En las tablas 8, 13 y 18, sitúa todas las figuras móviles en la zona no visible de la pantalla, en los planos de proyección, y más concretamente en la fila 209; sitúa a cada plano de proyección con la figura móvil cuyo número sea el mismo de ese plano de proyección, y todas las figuras móviles adquieren el color de fondo. En las tablas 9,14 y 19 el efecto es totalmente destructivo: llena de ceros toda la tabla, convirtiendo a todas las figuras móviles en transparentes.

La subrutina &H006C prepara la memoria de vídeo para el modo *SCREEN* 0, inicialmente las tablas 0 y 2, y los registros *VDP.* Recibe el nombre de *INITXT.*

La subrutina &H006F prepara la memoria de vídeo para el modo *SCREEN* 1 y, al igual que la subrutina anterior, inicializa todas las tablas de este modo de texto. Recibe el nombre de *INIT32.*

La subrutina &H0072, al igual que las dos anteriores, prepara la memoria de vídeo e inicializa las tablas del modo *SCREEN* 2 y, también como las dos anteriores prepara los registros *VDP* para este modo de gráficos. Recibe el nombre de *INIGRP.*

La subrutina &H0075, hace exactamente lo mismo que las tres anteriores, pero referida al modo *SCREEN* 3. Recibe el nombre de *INIMLT.*

La subrutina &H0078 es la primera de otras cuatro, también referidas a los cuatro modos *SCREEN,* pero de forma más limitada. Esta subrutina sólo hace parte del trabajo que hacía la subrutina &H006C, es decir, únicamente inicializa los registros *VDP* para el modo *SCREEN* 0, sin alterar las tablas de la memoria de vídeo. Esta subrutina recibe el nombre de *SETTXT.*

Las tres subrutinas siguientes, las &H007B, &H007E y &H0081, reciben los nombres de *SETT32, SETGRP* y *SETMLT,* respectivamente; sirven para inicializar los registros *VDP* de los modos *SCREEN* 1,2 y 3. Estas cuatro subrutinas no sirven para cambiar a un modo *SCREEN* completamente, sólo lo hacen parcialmente.

La subrutina siguiente no tiene nada que ver con las ocho anteriores. Es la número &H0084, recibe el nombre de *CALPAT,* y sirve para hallar el octeto de la memoria de vídeo correspondiente a la ta-

bla 9, 14 ó 19, según el modo, a partir del cual comienza el patrón de una figura móvil concreta. Si, por ejemplo, queremos hallar el patrón de la figura móvil que se halla en el plano de proyección 3, introducimos 3 en A, y tras ejecutar la subrutina, aparecerá en HL el octeto que buscamos.

La subrutina número &H0087, llamada *CALATR,* indica cuál es el octeto correspondiente a la coordenada «y» en las tablas 8,13 ó 18 de una figura móvil concreta. Al igual que la subrutina anterior, en A introducimos la figura móvil que queremos, y después en HL hallaremos el octeto que buscamos.

La subrutina número &H008A recibe el nombre de *GSPSI,* y sirve para averiguar si estamos trabajando con figuras móviles «grandes» (16 x 16) o «pequeñas» (8 x 8). Si son pequeñas, tras la subrutina en A aparecerá 8, y en el primer *bit* de F aparecerá 0. Si son grandes, en A aparecerá 32, y en el primer *bit* de F, 1.

La subrutina número &H008D recibe el nombre de *GRPPRT,* y sirve para pintar caracteres en la pantalla si os halláis en el modo *SCREEN* 2. Si introducís en A el código *ASCII* del carácter a escribir, ese carácter se situará en el lugar donde esté en ese momento el cursor de gráficos. (El cursor de gráficos es un cursor que puede moverse por la pantalla con otras subrutinas de las que hablamos a continuación.)

Las subrutinas siguientes del *BIOS* no tienen que ver con la memoria de vídeo hasta la &H00FC. Esta subrutina mueve el cursor de gráficos un punto hacia la derecha, y recibe el nombre de *RIGHTC.*

La subrutina &H00FF recibe el nombre de *LEFTC,* y hace exactamente lo contrario que la anterior: mueve un punto a la izquierda el cursor de gráficos.

La subrutina &H0102se llama *UPC,* y sirve para subir un punto el cursor de gráficos.

La subrutina siguiente es la &H0105, se llama *TUPC,* y es un caso especial de la anterior. Si aplicamos la subrutina anterior cuando el cursor de gráficos está en la línea superior, el cursor no se mueve, y no pasa nada, pero si ejecutamos *TUPC* y el cursor está en la línea superior, además de no moverse, sitúa un 1 en el primer bit del registro F. Por lo demás, *TUPC* actúa como *UPC* en todos los demás casos: siempre sube el cursor un punto.

Las subrutinas &H0108 y &H010B reciben, respectivamente, los nombres de *DOWNC* y *TDOWNC,* y son las equivalentes a las dos anteriores, pero en sentido opuesto. *DOWNC* baja un punto el cursor de gráficos y ya está, y *TDOWNC,* si el cursor está en la línea inferior, pone a 1 el bit de acarreo (el primer *bit* de F).



La subrutina &H010E, llamada *SCALXY,* sirve para averiguar si un punto concreto pertenece a la pantalla o no. En primer lugar se introduce la coordenada x del punto que queremos investigar en BC, y la coordenada y en DE. Tras la subrutina, si (x,y) pertenece a la pantalla, no cambia nada en BC y DE. Si alguna de las dos coordenadas (o ambas) son excesivamente grandes o excesivamente pequeñas, se les modifica de tal manera que la coordenada equivocada entre en la pantalla, si es por exceso en el punto máximo, y si es por defecto en el punto mínimo. Ejemplo: (–3,500) pasaría a ser (0,191). En el modo *SCREEN* 3 hay que aplicar las coordenadas del punto como si estuviéramos en el modo *SCREEN* 2, pues la transformación de esas coordenadas a nuevas coordenadas basadas en cuadrados de 4 x 4 *píxels,* las hace la misma subrutina.

Las subrutina &H0111 recibe también el nombre de *MAPXYC.* Sirve para que, una vez halladas las coordenadas correctas de nuestro punto mediante la subrutina anterior, averigüemos cuál es el octeto de la memoria de vídeo que controla ese punto. Poniendo la coordenada x en BC, la y en DE, y ejecutando la subrutina, hallaremos cuál es este octeto en hexadecimal, en los octetos &HF92A y &H92B de la *RAM.* Este octeto de la memoria de vídeo controla a ocho *píxels,* y para saber cuál de los ochos es el que buscamos, en el octeto &HF92C de la *RAM* aparecerá un número binario de ocho cifras, siete de las cuales serán 0 y

una de ellas 1. Ese 1 ocupará el lugar que ocupa nuestro píxel entre los otros siete de su mismo octeto.

La subrutina siguiente, la número &H0114, también llamada *FETCHC,* sirve para hallar el cursor de gráficos. Tras la ejecución de esta subrutina, en HL estará el octeto que controla al punto cursor y a otros siete, y A tendrá un número binario gracias al cual podremos hallar el orden del *píxel* cursor dentro de ese grupo de ocho *píxels.*

La subrutina número &H0117, llamada STOREC, es la contraria a la anterior, y permite fijar la posición del cursor. Para introducir su posición, tenemos que hacerlo igual que antes: en HL el octeto que lo controla, y en A el orden entre los otros siete *píxels.*

La subrutina &H011A sirve para establecer en el sistema operativo del ordenador cual ha de ser el color del cursor de gráficos. Para ello, tenéis que introducir en A el color que queráis que tenga, y a continuación ejecutar *CALL* &H011A. Esta subrutina también recibe el nombre de *SETATR.*

Si has ejecutado la subrutina &H0117, el ordenador ya sabe la posición del cursor de gráficos, y entonces podéis utilizar la subrutina &H011D, llamada *READC.* Sirve para hallar el color en el que se encuentra el cursor de gráficos. Tras la ejecución, el color aparecerá en A.

La subrutina &H0120, también llamada *SETC,* es una ampliación de la &H011A. Con la &H011A no se lograba poner el cursor de ese color, sólo preparaba el sistema operativo para que ese fuera el color del cursor. Con la *SETC* si se logra poner el cursor de un color concreto. Si el ordenador ya sabe la posición del cursor, se ejecuta &H011A, y a continuación &H0120, y el cursor adquirirá ese color.

La subrutina &H0123, llamada *NSETCX,* pinta de un color concreto unos cuantos puntos a la derecha del cursor. Si se emplea en el modo *SCREEN* 3, además el cusor se moverá un punto a la izquierda. Para usar esta subrutina, si el ordenador ya sabe la posición del cursor, sólo hay que introducir en HL la cantidad de puntos a pintar, y previamente utilizar &H011A para fijar el color del que queremos esos puntos.

La subrutina &H0126, llamada *GTASPC,* carga los octetos &HF40D y &HF40E de la *RAM* en HL y los octetos &HF40B y &HF40C en DE. Estos octetos sirven para expresar la relación altura-anchura en una elipse (que se puede trazar con *CIRCLE)* en función del radio. Esa relación en función del radio será un número concreto, a, y en los dos primeros octetos hay que introducir 256 x a, y en los dos segundos, 256/a.

La subrutina &H0129, también llamada *PNTINI,* sirve para el modo *SCREEN* 3, pues almacena el color del borde de una figura concreta en este modo *SCREEN.* Antes de la ejecución es necesario introducir en A el color del borde de la figura que queremos. Tras la ejecución, ese color estará guardado en el octeto &HFCB2 de la *RAM.* Si introducís un color, inecistente, el *bit* de acarreo de F se podrá a 0.

La subrutina &H012C, llamada *SCANR* es, como el anterior, muy conveniente si queréis utilizar *PAINT* desde Código Máquina. Esta subrutina, partiendo desde donde está el cursor, avanza hacia la derecha todos los puntos que puede, pintándolos todos de un color concreto, hasta encontrarse con un punto del color de borde de una figura determinada. El color de borde de la figura ya lo habréis tenido que establecer usando &H0129. El color del que va a pintar los puntos se establece usando &H011A, y en DE ponéis el número máximo de puntos que va a pintar el ordenador!

La subrutina &H012F, llamada *SCANL,* es exactamente igual que la anterior, pero en vez de avanzar hacia la derecha, avanza hacia la izquierda.

La última de estas subrutinas es la &H013E, llamada *RDVDP,* y almacena en A una copia del registro *VDP.*

La manera de usar el BIOS

Poco más hay que os podamos comentar acerca de estas subrutinas. Si queréis aprender a programar bien el Código Máquina, sólo lo conseguiréis programando muchas cosas, día tras día, pues la experiencia es lo que mejor os va a poder indicar si en un momento dado es mejor aplicar una subrutina u otra, y os va a poder enseñar muchos trucos que usaréis en futuras aplicaciones. Teniendo ya la lista de todas las subrutinas de la memoria de vídeo, y con los conocimientos adquiridos en esta serie, poseéis unas posibilidades ilimitadas para los mecanismos gráficos de vuestros programas.

...Y eso ha sido todo. No hay mucho más que se pueda decir sobre la memoria de vídeo. Suponemos que en poco tiempo nos comenzarán a llegar vuestros programas, muy evolucionados y cada vez mejores, capaces de aumentar la calidad de las ludotecas de nuestros lectores.

José M. Cavanillas

CURSO DE INGLES

The Gruneberg Linkword Language System es un sistema, para enseñanza de idiomas, más rápido y fácil que los métodos convencionales aplicados actualmente.
En poco tiempo, máximo 20 horas, te enseñará un vocabulario de 400 palabras y adquirirás unas buenas nociones de gramática. Esto te permitirá entender y ser entendido en tus viajes a lugares de habla inglesa o en tus contactos con personas que se expresen en ese idioma.
Por otra parte, el Sistema PlusData, consigue que el ordenador se convierta en un perfecto profesor que te explicará, orientará y corregirá, manteniendo en todo momento un "diálogo" interactivo de resultados sorprendentes.





Programas de EAO para EGB.
Cursos de Basic, Cobol, etc. AUTODIDACTAS.

Nombre ....
Apellidos ....
Dirección ....
Población ....
D.P. ....Tlno. ....
Forma de pago: Reembolso ☐ Giro postal ☐ Envío talón ☐

☐ Curso de Inglés 1.ª parte. 10 lecciones Linkword. (Cinta) P.V.P. 6.900.-Ptas.
☐ Curso de Inglés 1.ª parte. 10 lecciones Linkword. (3,5"-Disk) P.V.P. 7.900.-Ptas.
ENVIAR ESTE CUPON A: PLUS DATA, S.A. C/. GRAN VIA, 661 pral. 08010-Barcelona. Tel. 246 02 02

Spectravideo 318/328 SCREEN 2.

Gráficos de baja resolución

La principal característica del modo de baja resolución es que la unidad mínima de representación en la pantalla va a ser un conjunto de cuatro por cuatro *Píxels* al que pondremos el nombre de «macropunto».

En la tabla de la figura 3 que vimos en el capítulo II sobre el *VDP* podemos comprobar que son únicamente tres los registros que varían su contenido al cambiar de 1 a 2 el modo de pantalla.

En los registros 0 y 1 el cambio se debe a que se baja o pone a cero el *bit* que indica modo 1 de pantalla y se alza el que indica modo 2 (ver figuras 2 y 3 en los capítulos 1 y 2 sobre el *VDP*).

El registro 2 ha variado su contenido de 6 a 2 por lo que ahora la tabla de denominaciones empieza en la dirección 800 hex de la *VRAM* o memoria de vídeo.

El registro 3 no se usa en este modo de pantalla. En el capítulo anterior dedicado a la pantalla de alta resolución, vimos como este registro contenía la dirección donde empezaba la tabla de color que, más adelante veremos por que, no se usa en este modo de pantalla.

Los registros 4, 5 y 6 no han variado su contenido, lo cual indica que las tablas de *sprites* y la de generación de patrones no han variado su dirección de comienzo.

Los *sprites* y sus tablas ya los vimos en el capítulo anterior dedicado al modo de alta resolución. En la pantalla de baja resolución lo referente a *sprites* es idéntico a lo visto en dicho capítulo, por lo que aquí sólo apuntaremos el hecho de las grandes posibilidades que ofrece el poder manejar *sprites* de alta resolución en esta pantalla de baja resolución.

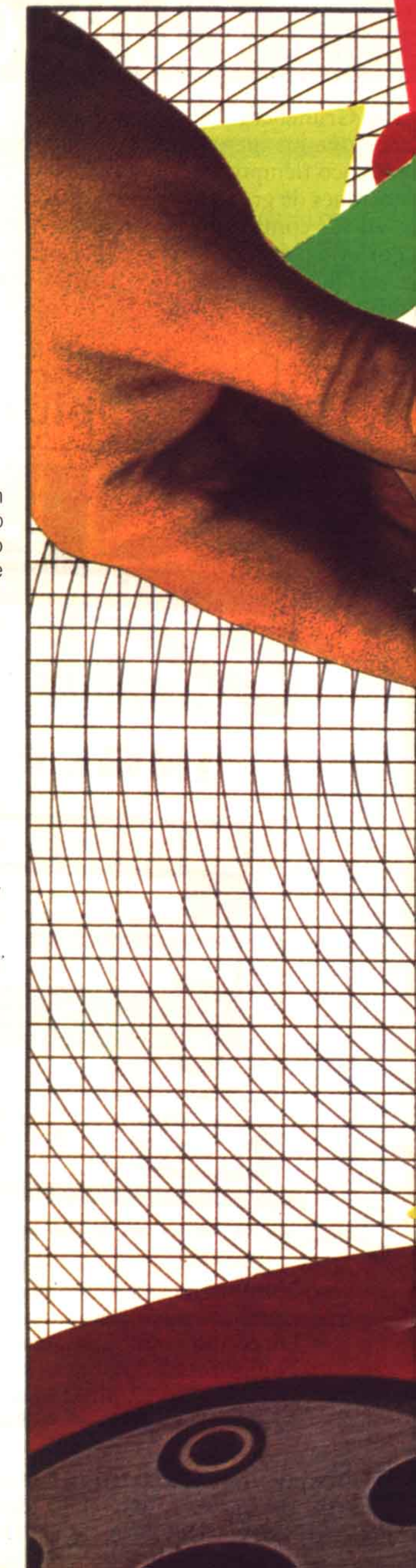
La tabla de generación de patrones

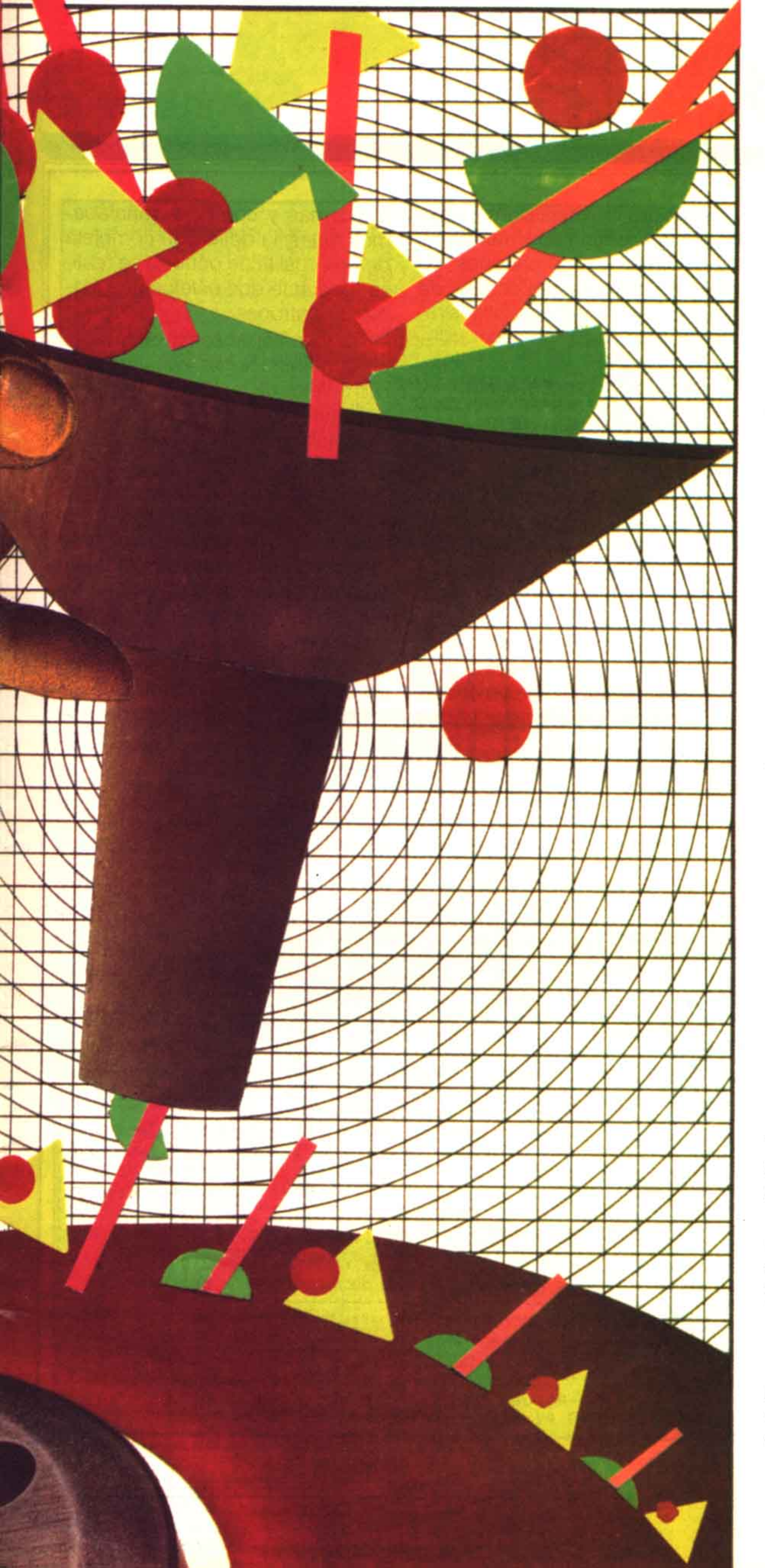
En este modo de pantalla la tabla de generación de patrones empieza, al igual que en este modo 1, en la dirección 0 de la *VRAM* o memoria de vídeo, pero su tamaño ha disminuido enormemente ya que ha pasado de ocupar 6 *kbytes* a ocupar 1,5 *Kbytes* de *VRAM,* (comparar mapas de *VRAM* del artículo del mes pasado y de éste).

Al comienzo del artículo hemos empezado a hablar de los macropuntos, vamos a ver qué relación tienen con la tabla de patrones.

Un macropunto siempre coincidirá con la cuarta parte de una casilla de la pantalla, por lo que cada una de ellas estará formada de cuatro macropuntos.

En la tabla de patrones se almacena el color que tiene cada uno de estos macropuntos que componen la pantalla de baja resolución. Esta es la razón por la que en este modo de pantalla no hace falta tabla de color.





En nuestro ordenador los códigos de colores están comprendidos entre 0 y 15, que son números factibles de representar con un grupo de cuatro *bits* o cuarteto.

Cada octeto de la tabla de patrones que estamos viendo contiene en cada una de sus mitades un número entre 0 y 15 que indica el color de un macropunto de la pantalla.

Así tenemos que en cada octeto de la tabla de generación de patrones estará contenido el color de dos macropuntos de la pantalla, que serán la mitad superior o la mitad inferior de una casilla de la misma. Es evidente que para definir una casilla en baja resolución harán falta dos octetos de la tabla de patrones.

Al igual que en la pantalla de alta resolución, no cualquier definición patrón de la tabla de patrones puede aparecer en una determinada posición de la pantalla, sino que depende de la fila en la que se encuentre dicha posición.

En alta resolución vimos como la pantalla se dividía en tres tercios y a cada tercio se la asignaba una parte de la tabla de patrones. Algo parecido ocurre en baja resolución, pero la forma de dividir la pantalla se efectúa de forma diferente.

Ahora en lugar de tres hay cuatro zonas, y las filas de pantalla que componen dichas zonas aparecen entremezcladas.

La primera zona está formada por las filas 0, 4, 8, 12, 16 y 20; la segunda por las filas 1, 5, 9, 13, 17 y 21, la tercera por la 2, 6, 10, 14, 18 y 22; la cuarta por la 3, 7, 11, 15, 19 y 23.

Los octetos que en la tabla de patrones corresponden a cada zona también aparecen entremezclados de la siguiente manera:

Los dos primeros octetos de la

tabla definirán los colores de la casilla de la primera zona cuyo octeto correspondiente de la tabla de denominaciones contega un 0, o podríamos decir también que definen el patrón 0 de la primera zona; los dos siguientes octetos definirán el patrón 0 de la segunda zona en que hemos dividido la pantalla, los dos siguientes el patrón 0 de la tercera y los dos restantes octetos del primer grupo de ocho definirán el patrón 0 de la cuarta zona. A continuación viene otro grupo de ocho octetos que contiene el patrón 1 de todas las zonas, el grupo de ocho octetos siguientes contiene los patrones 2, y así hasta el patrón 191.

Para darnos cuenta mejor de todo esto vamos a enviar unos valores a la tabla de patrones y vosotros mismos podréis apreciar los resultados.

```
10 SCREEN2
20 FOR A=0TO&h5FF:VPOKEA,A\6
30 FOR B=0TO50:NEXT:NEXT
40 GOTO40
```

La tabla de denominaciones

La tabla de denominaciones en el modo 2 de pantalla tiene el mismo tamaño y distribución que en el modo 1, esto es normal ya que, al igual que en dicho modo, hay un octeto por cada una de las casillas de pantalla, y hay 32*24 casillas.

En este modo la tabla de denominaciones comienza en la dirección 800 hex a diferencia del modo 1 en el que empezaba en la dirección 1800 hex.

Como hemos dicho antes, la correspondencia con la pantalla en el orden de situación de cada octeto en la tabla, es idéntica a la del modo 1, así a la primera casilla de la pantalla le corresponde el primer octeto de la tabla, a la segunda el segundo, etc.

Más claro lo veréis si probáis el siguiente ejemplo que introduce unos valores en los patrones iniciales de las primeras casillas de la pantalla y a continuación los repite en toda ella.

```
10 SCREEN 2
20 FORA=0TO7:VPOKEA,A:NEXT
30 FORA=&H800TO&HAFF:VPOKEA,0
40 FORB=0TO50:NEXT:NEXT
50 GOTO50
```

Si os fijáis en el ejemplo podréis apreciar que la pantalla aparece dividida en seis grandes franjas. Esto es así porque, como ya hemos visto al hablar de la tabla de patrones, dependiendo de la fila de pantalla que tratemos, el contenido de los octetos de la tabla de denominaciones se referirá a un patrón contenido en una u otra de las zonas de la tabla de patrones.

Además y como ya señalábamos antes, la definición completa de una casilla de pantalla se realiza mediante dos octetos de la tabla de patrones.

Por lo tanto, un octeto de la tabla de denominaciones señalará que dos octetos de la tabla de patrones definirán lo que ha de aparecer en la casilla correspondiente de la pantalla.

Para saber en que dirección de la *VRAM* se encuentran los dos octetos de la tabla de patrones correspondientes a una casilla de la pantalla a partir de la fila en que se encuentra y del número contenido en el octeto de la tabla de denominaciones correspondientes, se ha de aplicar la fórmula siguiente:

DIRECCION = Número de de-

DISTRIBUCION DE LA VRAM EN EL MODO 2

DIR. HEX.	CONTENIDO
0000- 5FF	TABLA DE GENERACION DE PATRONES --A cada casilla de la pantalla le corresponden dos octetos, el primero para la mitad superior y el segundo para la inferior. Bits 0 a 3 -- color del cuarto derecho Bits 4 a 7 -- color del cuarto izquierdo
600- 7FF	NO UTILIZADO
800- AFF	TABLA DE DENOMINACIONES --El número contenido en cada octeto ha de estar comprendido entre 0 y 191. Direcciónes patrón señaladas = número contenido * 8 + (fila de pantalla en que está MOD 4) *2 y la siguiente
B00-1AFF	NO UTILIZADO
1B00 1B7F	TABLA DE ATRIBUTOS DE SPRITES Cuatro bytes por cada uno de los 32 planos. 2 de coordenadas, 1 de tamaño y 1 de color.
1B80-37FF	NO UTILIZADO
3800 3FFF	TABLA DE GENERACION DE PATRONES DE SPRITES Se pueden representar 256 figuras de 8*8 pixels o 64 figuras de 16*16 pixels

```
 10 REM Dibujo de la figura inicial en pantalla
 20 COLOR1,15,1:SCREEN 2
 30 LINE(112,32)-(132,80),14,BF
 40 LINE(108,44)-(120,68),13,BF
 50 PSET(108,24),2
 60 DRAW"l4d4r16c13r4u4l4bd64br4l4u4r8c2r12d4l4"
 70 VPOKE394,&H88:VPOKE395,&H88
 80 VPOKE617,&H8D:VPOKE618,&H88
 90 VPOKE367,&H9D:VPOKE365,&H9D
100 VPOKE372,&H2F:VPOKE373,&H4B
110 VPOKE374,&H33:VPOKE375,&HB4
120 VPOKE624,&HF2
130 VPOKE370,&HEC:VPOKE626,&HEC
650 GOTO650
```

nominaciones * 8 + (fila MOD 4) * 2.

Si queréis partir directamente de la posición de pantalla debéis sustituir el número de denominaciones de la fórmula anterior como sigue:

Número de denominación =

```
140 REM Pasa figura a dirección 3000 hex de la VRAM
150 FORA=0TO&H5FF:VPOKEA+&H3000,VPEEK(A):NEXT
160 REM Cambio de la figura inicial
170 VPOKE110,&HFF:VPOKE111,&HFF
180 VPOKE118,&H22:VPOKE646,&H22
190 VPOKE653,&HFF:VPOKE654,&HFF
200 VPOKE394,&HFF:VPOKE395,&HFF
210 VPOKE617,&HFD:VPOKE618,&HFF
220 VPOKE367,&HAD:VPOKE365,&HAD
230 VPOKE372,&H42:VPOKE373,&H3F
240 VPOKE374,&HBB:VPOKE375,&HF3
250 VPOKE624,&H24
260 REM Pasa nueva figura a la dirección 2000 hex
270 FORA=0TO&H5FF:VPOKEA+&H2000,VPEEK(A):NEXT
280 REM Cambia de nuevo la figura anterior
290 VPOKE118,&HFF:VPOKE119,&HFF
300 VPOKE134,&H22:VPOKE135,&H22
310 VPOKE645,&HFF:VPOKE646,&HFF
320 VPOKE629,&H22:VPOKE630,&H22
330 VPOKE367,&H9D:VPOKE365,&H9D
340 VPOKE372,&H34:VPOKE373,&HB2
350 VPOKE374,&HFF:VPOKE375,&H2B
360 VPOKE624,&H43
370 REM Pasa la nueva figura a la dirección 1000 hex
380 FORA=0TO&H5FF:VPOKEA+&H1000,VPEEK(A):NEXT
390 REM Se cambia por última vez la figura
400 VPOKE134,&HFF:VPOKE630,&HFF
410 VPOKE142,&H22:VPOKE143,&H22
420 VPOKE621,&H22:VPOKE622,&H22
430 VPOKE362,&H88:VPOKE363,&H8D
440 VPOKE649,&H88:VPOKE650,&H88
450 VPOKE367,&HAD:VPOKE365,&HAD
460 VPOKE372,&HB3:VPOKE373,&HF4
470 VPOKE374,&H22:VPOKE375,&H4F
480 VPOKE624,&H3B
```

VPEEK (&H800 + 32 * número de fila + número de columna).

Los patrones de la casilla estarán contenidos en la dirección hallada y en la siguiente.

Algunas observaciones sobre el VDP

A lo largo de estos cuatro capítulos hemos ido viendo como el procesador de vídeo *TMS 9918 A* manejaba la imagen en cada uno de los tres modos de pantalla.

Vamos a ver ahora alguna característica que podemos sobresaltar del mismo.

En el registro 1 hemos visto que el *bit* 6 habilita y deshabilita la pantalla, lo cual nos permite realizar un dibujo con la pantalla deshabilitada con lo que no se verá mientras se está efectuando, para luego habilitar la pantalla con lo que el dibujo efectuado aparece de golpe en la misma.

Si os fijáis en los diferentes mapas de la memoria de vídeo a lo largo de estos capítulos notaréis que hay espacios vacíos en la asignación inicial que el sistema operativo hace de la misma.

Esto nos permite por ejemplo en el modo 0 introducir seis juegos más de caracteres además del inicial como ya vimos en un capítulo anterior.

En el modo 1 de pantalla el espacio no usado es muy pequeño ya que este es el modo de pantalla que mejor aprovecha la memoria de vídeo. No obstante, si quisiéramos crear más espacio libre en este modo en la memoria de vídeo podríamos usar el truco de asignar a varias de las casillas de la pantalla un patrón de la tabla de patrones común a todas ellas, me-

```
620  REM  Se conmutan inicios de la  tabla  de patrones
630  A=0:B=2:D=1:C=1
640  ONINTERVAL=10GOSUB660:INTERVALON
660  Z=INP(&H85):OUT&H81,A
670  OUT&H81,(4OR&H80)
680  A=A+C*2:C=C-2*(A=0)+2*(A=6)
710  RETURN
```

diante la alteración del contenido de los octetos de la tabla de patrones.

En el modo 2 de pantalla la tabla de patrones es cuatro veces

```
490 REM Repite figura en toda la Pantalla
500 FORA=0TO383:VPOKE&H800+384+A,VPEEK(&H800+A):NEXT
510 FORA=0TO23:VPOKE&H800+32*A,VPEEK(&H800+32*A+16)
515 VPOKE&H800+32*A+1,VPEEK(&H800+32*A+17)
516 VPOKE&H800+32*A+29,VPEEK(&H800+32*A+13)
520 VPOKE&H800+32*A+30,VPEEK(&H800+32*A+14)
525 VPOKE&H800+32*A+31,VPEEK(&H800+32*A+15)
530 FORB=0TO4:VPOKE&H800+32*A+B+5,VPEEK(&H800+32*A+B+13)
535 VPOKE&H800+32*A+B+21,VPEEK(&H800+32*A+B+13):NEXT
610 NEXT
```

mayor que en el modo 1 y además no hay tabla de color por lo que los espacios libres en la misma son mayores que los espacios ocupados.

Esto nos da la posibilidad de tener definidas en la memoria varias tablas de patrones o de denominaciones tanto de las figuras que aparecen en el plano 32 como de los *sprites*.

Esta característica se puede aprovechar para crear en *BASIC* figuras animadas, incluso pantallas animadas con una velocidad de animación propia del código máquina ya que para, por ejemplo, cambiar toda la pantalla basta con enviar un nuevo código a un determinado registro del generador de vídeo.

Vamos a ver esta posibilidad aplicada al modo 2 mediante un ejemplo.

```
539 REM Crea 7 nuevas tablas de nombres
540 FORB=0TO31:VPOKE&HC00+32*A+B,
     VPEEK(&H800+32*A+(B+1)*-(B<31)):NEXT
550 FORB=0TO31:VPOKE&H1800+32*A+B,
     VPEEK(&HC00+32*A+(B+1)*-(B<31)):NEXT
560 FORB=0TO31:VPOKE&H1C00+32*A+B,
     VPEEK(&H1800+32*A+(B+1)*-(B<31)):NEXT
570 FORB=0TO31:VPOKE&H2800+32*A+B,
     VPEEK(&H1C00+32*A+(B+1)*-(B<31)):NEXT
580 FORB=0TO31:VPOKE&H2C00+32*A+B,
     VPEEK(&H2800+32*A+(B+1)*-(B<31)):NEXT
590 FORB=0TO31:VPOKE&H3800+32*A+B,
     VPEEK(&H2C00+32*A+(B+1)*-(B<31)):NEXT
600 FORB=0TO31:VPOKE&H3C00+32*A+B,
     VPEEK(&H3800+32*A+(B+1)*-(B<31)):NEXT
685 REM Se conmutan inicios de tabla de denominaciones
690 OUT&H81,B:OUT&H81,(2OR&H80)
700 B=-(B+D)*(B<15)-2*(B>=15):D=DXOR2
710 RETURN
```

PROGRAMA 3

Desfile de la división robotizada (Programa comentado)

Primeramente dibujamos en la pantalla de baja resolución una figura que representa a un robot visto desde arriba, o sea en planta, utilizando el lenguaje de gráficos de nuestro ordenador y unos *POKES* a direcciones de la *VRAM*, según conveniencias.

Tras teclear estas líneas nos aparece en la pantalla nuestra figura inicial que vamos a intentar animar mediante la conmutación de diferentes bancos de la memoria de vídeo, para lo cual crearemos cuatro figuras con las cuatro fases de un movimiento y cada una de ellas la meteremos en una tabla de patrones en las localizaciones de la *VRAM* 3000 hex, 2000 hex, 1000 hex y 0000 hex.

Para ello incluimos en el programa anterior las siguientes líneas.

Y la última figura resultante se queda almacenada en el bloque que comienza en la dirección 0 hex.

Como hemos ido señalando en el programa, las líneas 150, 270 y 380 se encargan de ir almacenando las figuras creadas en sus respectivos lugares de almacenamiento.

La línea encargada de hacerlo copia íntegramente el contenido de la tabla de generación de patrones en una nueva tabla, de esta forma cualquier patrón definido inicialmente se copiará en una segunda dirección.

El proceso de traslado de las figuras entre las direcciones es un poco lento ya que son muchas pero lo interesante viene a la hora de ejecutar los movimientos como veremos a continuación, ya que lo único que tenemos que hacer una vez cada figura esté en su lugar es alterar el valor del registro 4.

Así podremos conmutar alternativamente las direcciones donde se encuentran los distintos patrones que definen las figuras con lo que lograremos un efecto de movimiento de la misma, mediante la introducción de las siguientes líneas, que se añaden a las anteriores.

Tras anãdir estas líneas a las anteriormente introducidas y tras arrancar el programa con *"RUN"* y esperar un momento a que se carguen las figuras en su sitio comprobaréis que la figura cobra movimiento.

Para variar su velocidad podéis variar en la línea 640 el número que está detrás del signo "=". Colocando un 1 se consigue la máxima velocidad que es bastante aceptable, y si no probar a hacerlo por el método de dibujar las figuras una a continuación de otra mediante la supresión de algunas líneas del programa y un *"GOTO"* en el lugar conveniente.

Ahora vamos a repetir la figura en toda la pantalla mediante la introducción en el lugar correspondiente de la tabla de nombres de los valores que indiquen el patrón diseñado.

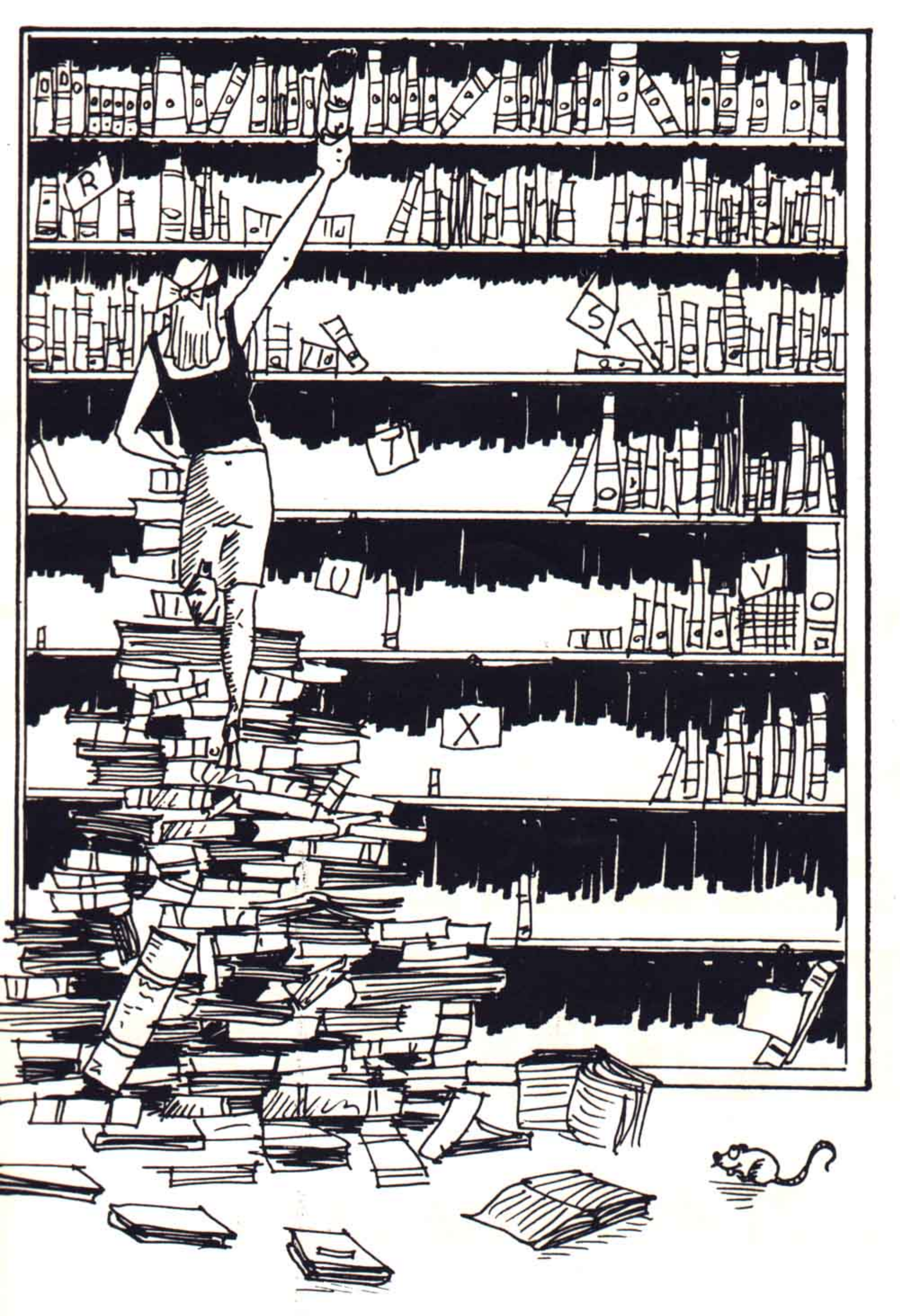
Si ahora pulsas *"RUN"* y *"ENTER"* verá que tras un momento de espera la figura se repite en toda la pantalla y a continuación se produce el movimiento de todas las figuras con la misma velocidad que lo hacía una sola de ellas.

Finalmente, vamos a hacer que las figuras además de cambiar de forma, se trasladen por la pantalla. Para ello vamos a crear siete nuevas tablas de denominaciones en la *VRAM,* que junto con la inicial formarán ocho diferentes grupos de denominación de las diferentes casillas de la pantalla, y a continuación además de variar el contenido del registro 4 variaremos también el registro 2 con lo que alteraremos el comienzo de la tabla de nombres.

Esta vez al teclear *"RUN"* la espera será un poco más larga pero el resultado vale la pena.

Venerando Solís

Biblioteca



Muchos de vosotros estaréis interesados en que algunos de los programas que se publican en **MSX Magazine** no sean una mera lista de sentencias, sino que estén suficientemente comentados. Pues bien, en estas páginas os vamos a contar cómo utilizar el programa que se lista junto a ellas y qué funciones realiza cada bloque del mismo.

El programa os permite almacenar una serie de fichas que contendrán los datos principales de los libros de vuestra biblioteca: título, autor, editorial y fecha de edición. Permite también que incluyáis en la ficha un breve comentario o notas. Los datos se almacenan en cinta, pero podéis alterar las rutinas correspondientes para utilizar la unidad de disco si disponéis de ella.

Se admite un bloque máximo de 200 fichas, lo que os permitirá disponer de unos 75 carácteres por ficha, por término medio. (Normalmente utilizaréis más de 70 carácteres por ficha, lo que reducirá el número de fichas disponibles). Si disponéis de una biblioteca más o menos amplia, os aconsejamos que abráis un fichero por cada tipo de libros, por ejemplo: poesía, novela, ciencias, etc. (Si tenéis unidad de disco, la única limitación es la capacidad del disco).

Una vez cargado el programa y pulsada la tecla *«run»*, aparecen las siguientes opciones en la pantalla. Todas ellas se activan pulsando una de las teclas de función (F1 a F8):

F1: Altas y Bajas.
F2: Listar Archivo.
F3: Grabar Archivo.
F4: Recuperar Archivo (de cinta).

F6: Accionar al Motor.
F7: Volver al menú principal.
F8: Finalizar (Interrumpir).

Vamos a comentar cada una de las opciones:

F1: ALTAS Y BAJAS. Pulsando F1 podéis almacenar nuevas fichas en el ordenador o bien eliminarlas.

Una vez pulsado F1, aparece en pantalla una nueva opción:

«A»: Pulsando la tecla A se abre una nueva ficha y podéis dar de alta un libro. El programa os pedirá entonces que le deis el título, después el autor, a continuación la editorial y por fin os permite hacer un comentario.

Por cada uno de estos conceptos se admite un máximo de 78 caracteres (dos líneas de pantalla). Para pasar al concepto siguiente, simplemente pulsar *RETURN.*

Observad que sólo se permite el uso de letras mayúsculas, los caracteres numéricos y algunos signos de puntuación. Esto se hace para una mayor facilidad de ordenación de las fichas y para mayor claridad de presentación. Por otra parte, las teclas del cursor son la tecla *RETURN* y la tecla *BS (Back Step)* que os permite retroceder en la línea que estáis editando si habéis cometido algún error.

Una vez hayáis introducido todos los datos, aparecerá en pantalla la ficha completa que acabáis de introducir. Para salir de aquí pulsar cualquier tecla y el ordenador volverá a pediros que pulséis «A» para altas, y «B» para bajas.

«B»: Al pulsar B podéis dar de baja una ficha, es decir, eliminarla. Entonces el ordenador os pedirá el número de la ficha que queréis eliminar. Una vez hayáis introducido el número, os preguntará si estáis seguros; si pulsáis «S» os preguntará si queréis que reordene el archivo (en el orden numérico); si volvéis a pulsar «S» lo hará así y si pulsáis «N» borrará la ficha, pero dejará el número libre (podréis volver a utilizarlo con la opción de modificación). En este caso, el resto de las fichas seguirán teniendo el número que tenían al principio. (Esto es útil, por ejemplo, si habéis colocado etiquetas a los libros con los números que les corresponden).

Para salir del modo F1: Altas y Bajas, podéis pulsar F7 para volver al menú principal o cualquiera de las teclas de función para realizar la operación correspondiente. Por ejemplo, si pulsáis F2 podéis listar el archivo y si pulsáis F3, grabarlo.

F2: LISTAR ARCHIVO. Esta es la opción que os permite realizar más cosas. Al pulsar F2 aparece en pantalla un segundo menú:

F1: Por orden alfabético.
F2: Alfabético, por títulos.
F3: Alfabético, por autores.
F4: Seleccionar ficha.
F5: Modificar ficha.

F6 y F8 quedan iguales que en el menú principal.

En cualquiera de las opciones F1, F2, F3 se lista el archivo completo; en el primer caso (F1) en orden numérico, en el segundo en orden alfabético por títulos y en el tercero en orden alfabético por autores. En todos los casos aparece una página con las primeras 20 fichas de la ordenación. Para pasar de página (avanzando o retrocediendo) utilizar las teclas del cursor. (Cursor arriba y cursor abajo). Estando en una de estas tres opciones se puede pasar a otra de ellas sin más que pulsar la tecla de función correspondiente. Además si pulsáis F4 podéis seleccionar una ficha simplemente dando el número. En este caso aparecerá en la pantalla dando el número. En este caso aparecerá en la pantalla la ficha completa. Si en lugar de F4 pulsáis F5, también seleccionamos una ficha, pero ahora podéis modificarla. El ordenador os preguntará qué es lo que deseáis modificar y posteriormente os permitirá hacerlo. Si habéis pulsado F5 por error, pulsar *RETURN* y no habrá modificaciones.

Estando en las opciones F4 y F5 podéis volver a listar el archivo pulsando F1, F2 o F3 según os interese.

Para volver al menú principal, como ya hemos indicado, basta pulsar F7.

F3: GRABAR ARCHIVO. Una vez hayáis terminado de dar de alta o baja las fichas correspondientes, os interesa grabar en cinta magnética el archivo completo. Para ello debéis pulsar F3. El programa empieza poniendo todas las fichas en orden numérico. A continuación os preguntará si deseáis introducir información. Si pulsáis «SI», esperará hasta que le deis dicha información (por ejemplo el tipo de libros de que se trata y la fecha) y pulséis «RETURN». Entonces se grabará el archivo completo en orden numérico.

Una vez terminada la grabación, el programa os pide que rebobinéis la cinta hasta el principio del archivo para comprobar que ha sido grabado correctamente. Si no ha sido así, os lo indicará con un mensaje y volverá a disponerse para grabar. Si la comprobación es correcta, vuelve al menú principal.

F4: RECUPERAR ARCHIVO. Cuando queráis tomar un archivo que está guardado en cinta, pulsad F4 a partir del menú principal.

De forma análoga a lo que ocurría con F3, el programa os preguntará si queréis información. Si pulsáis SI, tomará la información de la cinta y la presentará en pantalla.

Cuando se ha terminado de recuperar todo el archivo, el programa vuelve al menú principal y está listo para realizar cualquier operación: nuevas altas, modificaciones, etc.

Tanto en las opciones del menú principal como en las del menú de listado del archivo pueden realizarse tres interrupciones pulsando las teclas F6, F7 y F8:

F6: MOTOR. Acciona el motor de la grabadora, activándolo o desactivándolo según el estado en que esté.

F7: MENU. Pulsando F7 en cualquier momento, el programa se transfiere al menú principal.

F8: FIN. Al terminar de utilizar el programa, podéis teclear *CTRL STOP* o bien la tecla F8. En ambos casos el programa se interrumpe, la pantalla se limpia y se restablecen las teclas de función. También se restablece la tecla *CAPS LOCK* (mayúsculas) de forma que pueden escribirse minúsculas. (Como podréis observar, pulsar *CAPS LOCK* durante la ejecución del programa no sirve para nada: siempre está en mayúsculas).

Comentemos ahora, en grandes rasgos, lo que hace cada bloque del programa.

En la línea 60 se reserva memoria para las variables alfanumérica y para las rutinas en código máquina.

Las líneas 90 y 100 definen variables enteras (I, J, K) y alfanuméricas (A).

Entre las líneas 110 y 150 se almacena en memoria el DATA de la línea 3470. La rutina en *C/M* así definida se define en la línea 160 y sirve para encender la luz de la tecla *«CAPS LOCK»*. También se define (línea 170) una rutina del sistema que servirá para restablecer las teclas de función al interrumpir el programa.

Entre las líneas 180 y 230 se dimensiona la variable A que contendrá las fichas de la biblioteca; se define la variable AE que contiene los carácteres que se van a admitir y los encabezamientos (variable AC) que aparecerán en la pantalla a la entrada de datos o presentación de ficha.

En la línea 240 se activa la luz de la tecla *«CAPS LOCK»*, y mediante el *POKE* de la línea 250 se hace que sólo se admitan letras mayúsculas. Las líneas siguientes son bifurcaciones en caso de que se produzca *STOP* o *ERROR*.

A partir de la línea 330 y hasta la línea 560, se tiene el bloque correspondiente al menú principal. Cabe sólo señalar que con la línea 450 se escribe el mensaje "ARCHIVO COMPLETO" si se sobrepasa la dimensión de A o el espacio de memoria asignado a variables alfanuméricas. Si se pulsa una tecla que no sea de función, aparece el mensaje de la línea 510. En la línea 340 se llama a la subrutina (3240) que escribe en las teclas de función las operaciones que permiten realizar (con el DATA de la línea 3480).

Entre las líneas 610 y 1070 está el subprograma que permite dar fichas de alta y de baja. En el caso de altas, se llama a la subrutina de entradas (línea 3050 a 3190). Esta subrutina sólo admite los carácteres definidos en la variable AE (*INSTR* de la línea 3080). Sólo admite los códigos de control 13 *(RETURN)* para terminar la línea actual, y 8 *(BACK STEP)* que borra el carácter anterior, para corregir. En este caso se hace que la variable A que almacena la línea actual (líneas 3050 y 3150), pierda el último carácter (línea 3120) y, lógicamente, se borra el último carácter en la pantalla (línea 3110). Si la longitud de A llega a ser 78 (línea 3140) sólo se admiten los códigos 8 y 13.

Una vez se ha dado de alta la ficha completa, ésta se escribe en la pantalla mediante la subrutina de escritura de ficha (líneas 2640 a 2780) a la que se llama en la línea 750.

Cuando se da de baja una ficha, se muestra en la pantalla la ficha correspondiente. El programa permite (líneas 960 a 1060), eliminarla, reordenando el resto de las fichas.

Las rutinas principales para el listado están entre las líneas 1120 y 1840. Las líneas 1120 y 1340 presentan un menú similar al principal. Si se quiere un listado por orden numérico se ejecutan las líneas 1390 a 1420, si es por orden alfabético por títulos, las líneas 1430 a 1460, y si es por autores, las líneas 1470 a 1490. En todos los casos, si la variable IL no está activada, se realiza una ordenación de las fichas con la ordenación *SHELL-METZER* (líneas 1520 a 1670). El bloque comprendido entre las líneas 1710 y 1840 se encarga de hacer aparecer en la pantalla la página actual. Cada página tiene 20 líneas (variable JI). Sólo se admiten los códigos 30 y 31 (cursor) para pasar de página hacia delante o hacia atrás. Sólo se puede salir de este bloque pulsando una tecla de función.

Las líneas 1890 a 2310 permiten grabar el archivo. Primero se activa una bandera (variable AG en línea 1890) que permite ordenar el archivo en orden numérico

si no lo está ya (llamando a la subrutina de ordenación, línea 1400). Posteriormente se procede a la grabación sin más. Al final de la grabación se efectúa una comprobación de que ésta es correcta (líneas 2100 a 2310). Si no lo es, aparece un mensaje (líneas 2260 a 2310) y vuelve a la línea 1890 para volver a grabar. En caso contrario, se vuelve al menú principal (línea 2250).

La rutina de recuperación (líneas 2360 a 2530) es similar a la de grabación, con la única diferencia de que no se realiza la comprobación que se hacía allí.

Entre las líneas 2640 y 2780 está la rutina que permite presentar una ficha en pantalla. Como el archivo puede estar ordenado alfabéticamente, las líneas 2690 y 2700 se encargan de buscar que la ficha presentada tenga el número (A(I,O)) que se desea. Si se llama a esta rutina pulsando F4 desde el menú 2, sólo se puede salir de ella pulsando otra tecla de función. Los subprogramas correspondientes a altas y bajas y a modificaciones llaman a esta rutina.

La rutina de modificaciones corresponde a las líneas 2830 a 3000 y llama en primer lugar a la rutina de presentación de ficha (línea 2850). La variable AM de la línea 2880 contendrá las iniciales (TAEC) de los parámetros que se desea modificar. Esta variable se comprueba en 2940 y por medio de las líneas 2960 a 2980 se permite su modificación. (En 2970 se llama a la rutina de entrada de datos).

Entre las líneas 3360 y 3410 están las sentencias que se ejecutarán al provocar una interrupción del programa, pulsando «F8» o «*CTRL STOP*». Con la llamada de la línea 3360 se restablecen las teclas de función tal y como estaban al conectar el ordenador. El *POKE* de la línea 3390 hace que podamos poner minúsculas al pulsar «*CAPS LOCK*». Si después de interrumpir se pulsa *CONT* («F8»), el control del programa vuelve a la línea 240.

En 3420 se chequea si el error producido es por desbordamiento. La línea 3440 activa el motor (al pulsar «F6») y la línea 3460 vuelve al menú principal, al pulsar «F7».

J. Antonio Feberero

```
10 'ARCHIVO DE LIBROS
20 '=================
30 'Juan Antonio Feberero Castejón
40 'Versión 11.301185 - 6003 bytes
50 '
60 CLEAR 15000,59999!
70 SCREEN 0,,0
80 WIDTH 39
90 DEFINT I,J,N
100 DEFSTR A
110 RESTORE 3470
120 FOR I=1 TO 6
130 READ A
140 POKE 59999!+I,VAL("&H"+A)
150 NEXT I
160 DEFUSR0=&HEA60
170 DEFUSR1=&H3E
180 DIM A(200,4)
190 AE=CHR$(8)+"1234567890ABCDEFGH
     IJKLMNÑOPQRSTUVWXYZ.,;:'-=+\/
     ¡!¿? "
200 AC(1)="TITULO..........."
210 AC(2)="AUTOR............"
220 AC(3)="EDIT/LUGAR/AÑO..."
230 AC(4)="COMENTARIOS......"
240 U=USR0(0)
250 POKE &HFCAB,1
260 ON STOP GOSUB 3360
270 STOP ON
280 ON ERROR GOTO 3420
290 '
300 'MENU 1
310 '======
320 '
330 CLS
340 GOSUB 3240
350 LOCATE 11,3
360 PRINT "ARCHIVO DE LIBROS"
370 PRINT STRING$(39,192)
380 LOCATE 8,6
390 PRINT "F1 : ALTAS Y BAJAS"
400 PRINT TAB(8);"F2 : LISTAR ARCH
     IVO"
410 PRINT TAB(8);"F3 : [illegible]AR ARCH
     IVO"
420 PRINT TAB(8);"F4 : RECUPERAR A
     RCHIVO"
430 LOCATE ,15
440 PRINT "     Fulsar teclas de fu
     nción 1 a 10       para   re
     alizar las operaciones
      correspondientes."
450 IF IN=200 OR ER%=1 THEN LOCA
     ,21:PRINT TAB(8);"¡¡¡ARCHIVO
     COMPLETO!!!":ER%=0
460 ON KEY GOSUB 610,1120,1890,236
```

```
   0,,3440,3460,3360
470 KEY(1) ON:KEY(2) ON:KEY(3) ON:
    KEY(4) ON:KEY(6) ON:KEY(7) ON
    :KEY(8) ON
480 IF INKEY$="" THEN 480
490 BEEP
500 LOCATE ,21
510 PRINT "¡¡PULSAR SOLO TECLAS DE
     FUNCION!!"
520 FOR I=1 TO 1000
530 NEXT I
540 LOCATE ,21
550 PRINT SPACE$(33)
560 GOTO 480
570 '
580 'ALTAS Y BAJAS
590 '=============
600 '
610 CLS
620 IF IN=200 THEN 330
630 PRINT"<A>=ALTAS   <B>=BAJAS","
    Pulsa una tecla"
640 W$=INKEY$:IF W$="" THEN 640
650 IF W$<>"A" THEN 790
660 CLS
670 IN=IN+1
680 A(IN,0)=STR$(IN)
690 FOR I=1 TO 4
700 PRINT AC(I);"?"
710 GOSUB 3050
720 A(IN,I)=A
730 NEXT I
740 N=IN
750 GOSUB 2710
760 IF INKEY$="" THEN 760
770 CLS
780 GOTO 610
790 IF W$<>"B" THEN 640
800 CLS
810 INPUT"BAJA DEL LIBRO NUMERO...
    ";NF
820 GOSUB 2690
830 PRINT
840 PRINT
850 PRINT "¿DE ACUERDO...S/N?"
860 W$=INPUT$(1)
870 IF W$<>"S" THEN 610
880 PRINT "¿Deseas reordenación nu
     mérica...S/N?"
890 W$=INPUT$(1)
900 IF W$="S" THEN 960
910 IF W$<>"N" THEN 890
920 FOR J=1 TO 4
930 A(N,J)=""
940 NEXT J
950 GOTO 610
960 AG="GRAB"
970 GOSUB 1410
980 FOR I=NF TO IN-1
990 FOR J=1 TO 4
1000 A(I,J)=A(I+1,J)
1010 NEXT J
1020 NEXT I
1030 IN=IN-1
1040 IL=4
1050 GOSUB 1410
1060 AG=""
1070 GOTO 610
1080 '
1090 'MENU 2
1100 '======
1110 '
1120 CLS
1130 GOSUB 3260
1140 LOCATE 11,3:PRINT"CLASIFICACI
     ON:"
1150 PRINT STRING$(39,192)
1160 LOCATE 8,6
1170 PRINT "F1 : NUMERICA"
1180 PRINT TAB(8);"F2 : ALFABETICA
     -TITULOS"
1190 PRINT TAB(8);"F3 : ALFABETICA
     -AUTORES"
1200 PRINT TAB(8);"F4 : SEELCCIONA
     R FICHA"
1210 PRINT TAB(8);"F5 : MODIFICAR
     FICHA"
1220 LOCATE ,15
1230 PRINT "    Pulsar teclas de f
     unción 1 a 10        para    r
     ealizar las operaciones
       correspondientes."
1240 ON KEY GOSUB 1390,1430,1470,2
     640,2830,3440,3460,3360
1250 KEY(1) ON:KEY(2) ON:KEY(3) ON
     :KEY(4) ON:KEY(5) ON:KEY(6) O
     N:KEY(7) ON:KEY(8) ON
1260 IF INKEY$="" THEN 1260
1270 BEEP
1280 LOCATE ,21
1290 PRINT "¡¡PULSAR SOLO TECLAS D
     E FUNCION!!"
1300 FOR I=1 TO 1000
1310 NEXT I
1320 LOCATE ,21
1330 PRINT SPACE$(33)
1340 GOTO 1260
1350 '
1360 'LISTADO
1370 '=======
1380 '
1390 KEY(1) ON
1400 CLS
1410 IF IL=0 THEN 1680 ELSE IL=0
1420 GOTO 1500
```

```
1430 KEY(2) ON
1440 CLS
1450 IF IL=1 THEN 1680 ELSE IL=1
1460 GOTO 1500
1470 KEY(3) ON
1480 CLS
1490 IF IL=2 THEN 1680 ELSE IL=2
1500 KEY(4) STOP
1510 KEY(5) STOP
1520 JP=IN
1530 JP=JP\2
1540 IF JP<1 THEN 1680
1550 JN=1
1560 JK=IN-JP
1570 JM=JN
1580 JH=JM+JP
1590 IF IL=0 THEN IF VAL(A(JM,0))<
     =VAL(A(JH,0)) THEN 1660 ELSE
     1610
1600 IF A(JM,IL)<=A(JH,IL) THEN 16
     60
1610 FOR I=0 TO 4
1620 SWAP A(JM,I),A(JH,I)
1630 NEXT I
1640 JM=JM-JP
1650 IF JM<1 THEN 1660 ELSE 1580
1660 JN=JN+1
1670 IF JN>JK THEN 1530 ELSE 1570
1680 KEY(4) ON
1690 KEY(5) ON
1700 IF AG="GRAB" THEN RETURN
1710 J1=1
1720 IF IL=0 THEN IM=1 ELSE IM=IL
1730 FOR J=1 TO IN
1740 PRINT USING"### ";VAL(A(J,0))
     ;
1750 PRINT LEFT$(A(J,IM),30)
1760 IF J=IN THEN 1780
1770 IF J1<20 THEN J1=J1+1:GOTO 18
     40
1780 W$=INKEY$:IF W$="" THEN 1780
1790 IF IN<=20 THEN 1780
1800 IF ASC(W$)=31 AND J<IN THEN 1
     830
1810 IF ASC(W$)=30 AND J>20 THEN J
     =J-20-J1:GOTO 1830
1820 GOTO 1780
1830 J1=1:CLS
1840 NEXT J
1850 '
1860 'GRABACION
1870 '=========
1880 '
1890 AG="GRAB"
1900 A=""
1910 GOSUB 2540
1920 IF IL<>0 THEN GOSUB 1400
1930 OPEN"CAS:LIBROS" FOR OUTPUT A
     S#1
1940 LOCATE,5
1950 PRINT "¿Deseas grabar informa
     ción?"
1960 W$=INPUT$(2)
1970 IF W$="NO" THEN 2000
1980 PRINT "Introduce información"
1990 LINE INPUT A
2000 PRINT#1,A
2010 PRINT#1,IN
2020 FOR J=1 TO IN
2030 FOR I=0 TO 4
2040 PRINT#1,A(J,I)
2050 NEXT I
2060 NEXT J
2070 CLOSE#1
2080 CLS
2090 AG=""
2100 PRINT "Rebobina la cinta y pu
     lsa RETURN.        Voy a compro
     bar la grabación."
2110 IF INKEY$<>CHR$(13) THEN 2110
2120 PRINT "Comprobando..."
2130 OPEN"CAS:LIBROS" FOR INPUT AS
     #1
2140 LINE INPUT#1,AP
2150 IF AP<>A THEN 2260
2160 INPUT#1,IP
2170 IF IP<>IN THEN 2260
2180 FOR J=1 TO IN
2190 FOR I=0 TO 4
2200 LINE INPUT#1,AP
2210 IF AP<>A(J,I) THEN 2260
2220 NEXT I
2230 NEXT J
2240 CLOSE#1
2250 RETURN 330
2260 PRINT
2270 PRINT "¡¡¡ARCHIVO MAL GRABADO
     !!!"
2280 PRINT "    INTENTALO DE NUEVO.
     "
2290 PRINT "       PULSA UNA TECLA"
2300 IF INKEY$="" THEN 2300
2310 GOTO 1890
2320 '
2330 'RECUPERACION
2340 '============
2350 '
2360 GOSUB 2540
2370 OPEN"CAS:LIBROS" FOR INPUT AS
     #1
2380 LOCATE,5
2390 PRINT "¿Deseas recibir inform
     ación?"
2400 W$=INPUT$(2)
2410 LINE INPUT#1,A
```

```
2420 IF W$="NO" THEN 2440
2430 PRINT A
2440 INPUT#1,IN
2450 FOR J=1 TO IN
2460 IF EOF(1)=-1 THEN 2510
2470 FOR I=0 TO 4
2480 LINE INPUT#1,A(J,I)
2490 NEXT I
2500 NEXT J
2510 CLOSE#1
2520 IL=0
2530 RETURN 330
2540 CLS
2550 PRINT"Prepara la grabadora"
2560 PRINT"Pulsa [RETURN]"
2570 W$=INKEY$:IF W$="" THEN 2570
2580 IF ASC(W$)=13 THEN RETURN
2590 GOTO 2570
2600 '
2610 'FICHA
2620 '=====
2630 '
2640 LOCATE 0,22
2650 INPUT "Pulsa nº + RETURN";NF
2660 GOSUB 2690
2670 IF INKEY$="" THEN 2670
2680 RETURN
2690 FOR N=1 TO IN
2700 IF VAL(A(N,0))<>NF THEN NEXT
     N
2710 CLS
2720 PRINT "Nº:";A(N,0)
2730 FOR I=1 TO 4
2740 PRINT
2750 PRINT AC(I)
2760 PRINT A(N,I)
2770 NEXT I
2780 RETURN
2790 '
2800 'MODIFICACIONES
2810 '==============
2820 '
2830 LOCATE0,21
2840 INPUT "Pulsa nº + RETURN";NF
2850 GOSUB 2690
2860 PRINT
2870 AM=""
2880 INPUT "Pulsa inicial(es) de l
     o que deseas     modificar (T
     AEC) sin  comas + RETURN   ";
     AM
2890 IF LEN(AM)<4 THEN AM=AM+" ":G
     OTO 2890
2900 CLS
2910 GOSUB 2690
2920 PRINT
2930 FOR NM=1 TO 4
2940 NI=INSTR("TAEC",MID$(AM,NM,1)
     )
2950 IF NI=0 THEN 2990
2960 PRINT AC(NI);"?"
2970 GOSUB 3050
2980 A(N,NI)=A
2990 NEXT NM
3000 GOTO 2660
3010 '
3020 'ENTRADAS
3030 '========
3040 '
3050 A="":LOCATE,,1
3060 W$=INKEY$:IF W$="" THEN 3060
3070 IF W$=CHR$(13) THEN 3180
3080 IF INSTR(AE,W$)=0 THEN 3060
3090 IF W$<>CHR$(8) THEN 3140
3100 IF LEN(A)=0 THEN 3060
3110 PRINT W$;" ";W$;
3120 A=LEFT$(A,LEN(A)-1)
3130 GOTO 3060
3140 IF LEN(A)>=78 THEN 3060
3150 A=A+W$
3160 PRINT W$;
3170 GOTO 3060
3180 LOCATE0,CSRLIN+1,0
3190 RETURN
3200 '
3210 'Def. teclas de función
3220 '----------------------
3230 '
3240 RESTORE 3480
3250 GOTO 3270
3260 RESTORE 3490
3270 FOR I=1 TO 10
3280 READ A
3290 KEY I,A
3300 NEXT I
3310 RETURN
3320 '
3330 'Interrupciones
3340 '--------------
3350 '
3360 U=USR1(0)
3370 CLS
3380 LOCATE ,,0
3390 POKE &HFCAB,255
3400 STOP
3410 GOTO 240
3420 IF ERR=14 THEN ER%=1
3430 RESUME 330
3440 MOTOR
3450 RETURN
3460 RETURN 330
3470 DATA 3E,00,CD,32,01,C9
3480 DATA Alt/Baj,List A.,Grab A.,
     Rec A.,,Motor,Menú,Fin,,
3490 DATA Num.,Titulos,Autores,Sel
     ecc,Modif.,Motor,Menú,Fin,,
```

compro, vendo, cambio . . .

• INTERCAMBIO juegos de cartucho. Interesados escribir a: Juan José Ritoré. García de Paredes, 19. 06006 Badajoz.

• CAMBIO/VENDO programas. Escribir a: Luis A. de la Fuente. Gran Capitán, 3-5 2.° Derecha. 37005 Salamanca.

• VENDO ordenador SVI-328 con 605 Expander + programas (Wordstar, Turbo Pascal, Cobol, Forth, Fortran 80 y juegos) + cable Centronics. Todo por 70.000 ptas. (o mandar ofertas). Escribir a: Salvador Martínez. Peñalara, 23, 2.° A. Tres Cantos (Madrid).

• INTERCAMBIO programas. Escribir a: Javier Cabrera López, Z.R. Los Angeles, 1, 4.° A. Estepona (Málaga).

• CAMBIO/VENDO programas para MSX. Escribir a: Miguel Moreno. Cataluña, 48. El Prat de Llobregat. 08820 Barcelona.

• VENDO enciclopedia de la Informática BASIC (Ed. Forum), 6 tomos BASIC encuadernados, más 3 tomos BASIC Advanced. Todo al 60% del valor real. Impecable estado. Escribir a: Jesús Blanch. Central Telex Telégrafos. Plaza País Valenciano. 46002 Valencia.

• VENDO Spectravideo SV-328, cassette y revistas, por 40.000 ptas. Llamar a Montse Sala al Tel.: (93) 427 39 50.

• CAMBIO/VENDO toda clase de programas. Escribir a: Francisco Tomás Sánchez Marín. Río Segura, 11. 30600 Archena (Murcia).

• BUSCO la cinta-programa de la Tableta Gráfica del Spectravideo SVI-105. Escribir a: Carlos Martínez Martínez. Callejón del Moro, 1. Huete (Cuenca).

• VENDO ordenador Spectravideo 729 MSX nuevo. Incluyo Base de Datos, programas y libros. Todo por 50.000 ptas. Llamar a Francisco Gutiérrez Soto al Tel.: (91) 666 24 24.

• DESEO contactar con usuarios MSX y con toda clase de clubs para intercambiar programas. Interesados escribir a: Daniel Rincón Prada. Paseo San Raimundo, 41. Fitero (Navarra).

• DESEARIA contactar con usuarios del MSX. Escribir o telefonerar a: Angel Ramón Pérez Corzo. García Lorca, s/n. Arroyo de San Serván (Badajoz). Tel.: (924) 34 21 28.

ANUNCIOS GRATUITOS

Todos los anuncios (compras, ventas, cambios o comunicaciones de clubs de usuarios, etc.) que van en esta sección, deben tener un máximo de cuarenta palabras. Con el fin de facilitar la transcripción de los anuncios hemos recuadrado cuarenta espacios para que en cada uno vaya una palabra. Después, recortar y mandar a:

MSX MAGAZINE

ANUNCIOS GRATUITOS
C/ Bravo Murillo, 377, 5.° A
28020 MADRID

Rincón del lector

PUBLICACIONES SOBRE PROGRAMAS DE APLICACION

Soy propietario de un Toshiba HX-10 y me dirijo a ustedes con el fin de que me informen sobre libros y publicaciones sobre programas de contabilidad, archivos, facturación, etc.

J. Ernesto Prieto
Vigo

No existe mucha bibliografía sobre programas de gestión y especialmente dedicada al estándar MSX. En realidad, lo poco que hay en el mercado lo comercializan tres editoriales muy introducidas en este campo que son: Anaya Multimedia, Paraninfo y Ra-Ma.

Anaya tiene un amplio abanico de posibilidades, que incluso llega a tocar temas tan específicos como los referentes al IBM PC, pero dudamos que haya algún libro específico sobre programas de gestión para MSX. En cuanto a Paraninfo y Ra-Ma, no estamos muy seguros de que hayan publicado algo referente al tema. De cualquier manera, Ra-Ma que suele publicar libros muy interesantes (aunque los buenos están en inglés) puede tener el libro que buscas.

Las direcciones de las tres editoriales son las que a continuación te indicamos. Ellos, pueden orientarte mucho mejor, ya que están en constant renovación. En los catálogos que publican (no sabemos si mensual o bimensual) aparecen las últimas novedades, tanto en Inglaterra como en nuestro país.

Ra-Ma
Ctra. de Canillas, 144
28043 Madrid

Anaya Multimedia
Villafranca, 22
Tel. 245 82 05
28028 Madrid

Paraninfo, S.A.
Magallanes, 25
Tel. 446 33 50
28015 Madrid

EL SIMBOLO «#»

Me dirijo a ustedes para plantearles una simple y corta cuestión. ¿Cuál es el significado en el MSX del signo «#»? ya que el BASIC lo interpreta igual que otros caracteres.

Xavier Casajuana
Barcelona

El símbolo «#» no tiene ningún significado especial en el BASIC MSX. Simplemente es uno de los caracteres que no son interpretados por el BASIC, por lo que no produce un error cuando se teclea en modo directo y se pulsa la tecla RETURN.

CONTROLAR LA IMPRESORA

Tengo un Spectravídeo 728 y una impresora Star SQ-10, estoy intentando realizar un programa para escribir textos y me encuentro ante el problema de no saber cuándo la impresora se encuentra «ON LINE», es decir, preparada para recibir un carácter. Agradecería que me indicasen cómo puedo comprobar si la impresora está lista para recibir un carácter.

Joaquín S. Maza
Las Palmas de Gran Canaria

Aunque el BASIC no dispone de una función que permita comprobar el estado de la impresora, en el *BIOS* (rutinas utilizadas por el sistema operativo) existe una rutina que permite comprobar si la impresora está lista para recibir un carácter.

Esta rutina empieza en la dirección AB hexadecimal y puede ser llamada desde un pequeño programa en código máquina. Al retornar esta rutina, devolverá un 255 en el acumulador y un 0 en el flag Z si la impresora está preparada, en caso contrario, devolverá un 0 en el acumulador y un 1 en el flag Z.

MAPA DE MEMORIA Y VARIABLES DEL SISTEMA

Soy un asiduo lector de vuestra revista. Poseo un ordenador Spectravídeo 728 y aunque mi afición ha estado siempre enfocada al terreno de las aplicaciones, últimamente estoy verdaderamente interesado en programar vídeo juegos, pero mis posibilidades se ven ampliamente mermadas, ya que al no disponer de información sobre el sistema de paginación de memoria, no puedo almacenar una pantalla en SCREEN 2 directamente. Agradecería que me indicasen dónde puedo conseguir una lista de las variables del sistema y cómo almacenar dicha pantalla.

Francisco Martínez
Cádiz

Anaya Multimedia y Ferre Moret son las dos editoriales que publican los libros más prácticos del mercado. No tienen mucha paja y van directamente a las aplicaciones interesantes (como los de Ferre Moret).

En cuanto a la información que necesites sobre las variables del sistema, esta la hallarás en cualquiera de los libros siguientes:

—MSX, Guía del programador y manual de referencia (Ed. Anaya Multimedia).
—MSX, lenguaje máquina (Ed. Ferre Moret).
—MSX. Gráficos y sonidoi (Ed. Ferre Moret).

Salvar pantallas directamente no es posible, ya que la *VRAM* no está paginada como el resto de la memoria. Una buena solución es utilizar las rutinas del *BIOS* para trasladar la memoria de pantalla a una zona de la RAM y grabarla como código máquina.

SEIKOSHA

"IMPRESORAS PARA TODOS"

MP - 1300 "PARA TU PC"

- Impresora matricial con más de 200 tipos de letra y opción de color.
- 300 cps en standard, 64 cps en alta calidad.
- Velocidad de homologación 10.468 cpm al 100% y 2.549 cpm al 10%.
- Carro 10 pulgadas. **Mod. MP-5300 carro de 15 pulgadas**
- Tracción y fricción. Carga de papel posterior e inferior.
- Introductor automático de documentos hoja a hoja.
- Dos interfaces incluidas, paralelo centronics y RS-232.
- Buffer de 10K (7K con caracteres programables).
- Gran variedad de caracteres y gráficos.
- Dos modos de impresión: IBM y EPSON.
- Más de 256 caracteres programables.
- Fijación de márgenes en el panel frontal.
- Volcado de datos en hexadecimal.

Accesorios opcionales:
MP-13051 Cartucho de tinta negra.
MP-13055 Cartucho de tinta de cuatro colores.
MP-13009 Introductor automático de papel.
MP-13005 Kit de color.

MP-1300 - P.V.P. 119.900 Pts.
IVA NO INCLUIDO

MP-5300 - P.V.P. 149.900 Pts.
IVA NO INCLUIDO



BP - 5420 "PARA TU ORDENADOR"

- Impresora matricial con más de 150 tipos de letra.
- Tipos de letra seleccionados por Hard. y Soft.
- 420 cps en standard, 104 cps en alta calidad.
- Velocidad de homologación 20.104 cpm al 100% y 4.956 cpm al 10%.
- Máximo de carro 15 pulgadas.
- Dos modos de impresión: IBM y EPSON.
- Tracción y fricción. Carga de papel posterior e inferior.
- Volcado de datos en hexadecimal.
- Dos interfaces incluidas, paralelo centronics y RS-232.
- Buffer de 18K.
- Fiabilidad: Tiempo medio entre fallos 800 h.
- N° medio de caracteres entre fallos 200.000.000.

Accesorios opcionales:
BP-54051 Cartucho de tinta.
BP-CSF Introductor automático de papel.

P.V.P. 339.900 Pts.
IVA NO INCLUIDO

SP - 1000 "PARA TU MICRO"

- Matriz de impacto (9-pins)/10 pulgadas (Bidireccional optimizada).
- 100 cps en standard, 24 cps en alta calidad.
- Velocidad de homologación 4.339 cpm al 100% y 1.274 cpm al 10%.
- Gran variedad de tipos de caracteres.
- 96 caracteres en RAM, programables por el usuario. (del 32 al 127).
- Todos los tipos de letra definibles con un solo byte.
- Función de fijación de márgenes a derecha e izquierda.
- Tracción y fricción, introductor automático de papel hoja a hoja.
- Larga vida del cartucho de tinta.
- Compatible paralelo Centronics.
- Volcado de datos en hexadecimal.

P.V.P. 57.500 Pts.
IVA NO INCLUIDO

Accesorios opcionales:
SP-80051 Cartucho de tinta.
SP-80010 Interface serial.
SP-CS Introductor automático de documentos.

MODELOS SERIE SP
SP 1000 AS RS-232 versión serial.
SP 1000 VC Commodore compatible con C-64/VIC-20.
SP 1000 AP Apple II y Mac. Compatible con Macintosh.
SP 1000 MX Compatible con todos los ordenadores de norma MSX.
SP 1000 CPC Compatible con los ordenadores AMSTRAD.
SP 1000 I Compatible con IBM-PC.



Periféricos de Etiqueta

Blasco Ibáñez, 116 Tel. (96) 372 88 89 Telex 62220 DIRA E 46022-VALENCIA
Agustín de Foxá, 25-3°-A Tels. (91) 733 57 00-733 56 50 28036-MADRID
Muntaner, 60-2°-4ª Tel. (93) 323 32 19 08011-BARCELONA
Artazagone, 9 Tel. (94) 463 18 05 - LEJONA (Vizcaya)
Urbanización Mayber, 7 Tel. (922) 26 01 75 - Ctra. a Geneto LA LAGUNA (Tenerife)